Automatic Texture Classification in Manufactured Paper

Barnabas Ndlovu Gatsheni

A thesis submitted for the degree of Doctor of Philosophy.
The University of Edinburgh.
December 2001

Abstract

The automatic classification of manufactured paper must be seen as an integral part of the paper making industry. Currently the human element plays a pivotal role in the quality assessment of manufactured paper. This renders the inspection results unreliable as the human element is susceptible to different moods, social pressures and fatigue among others [1].

The system presented in this thesis replicates the actions of the human element in the quality assessment of manufactured paper and also expresses the subjective judgement for an objective figure of merit. This is achieved through the application of texture analysis in the characterisation of the surface appearance of paper for quality. However, texture analysis techniques individually give unsatisfactory classification performance. This thesis has shown that the use of multiple features from different techniques in combination leads to enhanced classification performance over the use of features from any single method alone.

Techniques from computer image analysis that were found useful for characterising the paper surface included the co-occurrence matrices, the grey level run length method, the specific perimeter method and first order statistics. A supervised neural network classifier was used for classification.

Confusion matrices and the loss matrices have been used for the first time for interpreting the paper classification results.

An intelligent feature selection strategy (intuition) has been found to be a powerful tool in paper classification. Furthermore, a combination of features from different techniques performed better than features from a single technique. A classification performance of 87% has been achieved on two classes of paper, the "good" and "poor" quality paper. The results suggest that classifying paper is a difficult problem.

This thesis has examined the suggestion that an automated paper classification system based upon multiple texture based features trained to match the performance of the human visual system.

Acknowledgements

I would like to thank my Supervior Prof Alan F. Murray for encouragement, patience and guidance. I will also like to thank my other supervisors Dr. Peter Edwards and Dr. David Renshaw for their guidance during the course of this work.

I feel also indebted to colleagues notably Andrew Peacock and Peter Hillman for checking on the robustness of some of the code in this vision system. I am also indebted to Shedden Masupe for continous encouragement.

From the department of artificial Intelligence (AI) in Edinburgh I will also like to thank Dr. Bob Fisher for discussions on co-occurrence matrices, Dr. Chris Williams for introducing me to loss matrices and the general area of data machine learning and probabilistic reasoning methods. I will also like to thank Dr. Anthony Ashbrook also from the AI department on discussions on the Gabor filters.

I will also like to thank Fabian Borocin, a colleague from the British Geological Survey, Edinburgh for useful discussions that we used to have on wavelets.

I wish to extend my gratitude to the Department of Electronics and Electrical Engineering of the University of Edinburgh for providing a Scholarship that supported this work. Without such support it would have not been possible to pursue this work. I will also like to acknowledge the assistance in funding for the difference between local and overseas fees by the National University of Science and Technology in Bulawayo.

I would like to dedicate this thesis to my parents who would have loved to live to see me complete this work.

Contents

List of figures

List of tables

Acronyms and abbreviations

SGLDM	Spatial dependence grey level matrices
GLDM	Grey Level Dependence Method
GLRLM	Grey Level Run Length Method
NGLDM	Neighbourhood Grey Level Dependence Matrix
DFT	Discrete Fourier Transform
DWT	Discrete Wavelet transform
DWF	Discrete Wavelet Frames
WFT	Windowed Fourier Transform
FFT	Fast Fourier Transform
FD	Fractal Dimension
HVS	Human Visual System
CD	Cross Direction
MD	Machine Direction
GMRF	Gaussian Markov Random Fields
MLE	Maximum Likelihood Estimates
SPM	Specific Perimeter Method
FOV	Field of View
CCD	Charged Coupled Devices
PIXEL	picture element
TEXEL	Texture Element
FOS	First order statistics
IDM	Inverse difference moment
DC	Direct current
AC	Alternating current
TSWT	Tree Structured Wavelet Transform
DCT	Discrete Cosine Transform
DBC	Differential Box Counting Method
GA	Genetic Algorithm
MLP	Multilayer perceptron
PCA	Principal Components Analysis
NN	Neural Networks

Chapter 1
Introduction

1.1 Introduction

This *thesis* examines the suggestion that an *automated paper-classification system* based upon multiple texture-based features and trained to emulate a human operator can approach the performance of the human visual system.

New advances in electronic technology have increased demand for high quality paper particularly in the printing industry. High quality paper means a uniform surface appearance. In assessing quality, local non-uniformities and an overall view of paper quality are taken into account. There is therefore need for paper making companies to invest in inspection systems.

Currently the human operator uses glancing angle illumination to view the surface appearance of the sample of paper in both the machine direction (the direction in the motion of the conveyor belt system) and the cross-direction (the direction in the same plane but perpendicular to the direction of movement of the conveyor system). The assessment of the surface appearance quality is normally performed "off-line". The size of features and their distribution on the paper surface are the key factors used in distinguishing between "good" and "poor" quality. Poor surface appearance can result from Fibre Furnish, Sheet Formation and Wet Felt Marking among other factors [2].

Data (sample) is collected every hour from the machine for inspection. The sample is then ranked according to perceived quality. However, the human inspector is not capable of putting a number to the grade of the sample. Secondly, inspection results might be variable where there is shift work. The assessment is therefore complex and subjective as it is based upon past experience and on a particular product's quality criteria. To maintain the quality of the product, any reduction in quality must be corrected immediately.

Operators (Human) are trained using examples of a range of qualities that form a scale. In an attempt to replace the operators, this thesis will develop a system that "learns" to mimic the subjective judgement of the operator and then categorise the samples. This vision system must use

computer image analysis and be based on texture in order to quantify what the eye can see. The texture characterisation process aims at producing accurate and repeatable results. In addition it must put a number to the quality of paper. The measures that will be used in modelling the proposed vision system must be able to describe the texture of the paper surface adequately. A suitable classification strategy will be adopted for discriminating between different categories of paper samples.

1.1.1 Motivation for the work

Despite major advances in automating papermaking, paper surface inspection has not been automated. The reason being that paper surface has to lose all the moisture before it can be inspected. The human visual system (HVS) upon which most paper industries rely, is adapted to variety and also requires observing the same type of image repeatedly to detect anomalies [3] [1]. Furthermore, the HVS is susceptible to boredom, stress, fatigue and is not repeatable. Additionally, the accuracy of the HVS in inspection declines with dull, routine jobs [4–6]. The HVS also struggles to discriminate between two textures that locally have the same second-order statistics [7]. Thus physiological limitations compromise the quality assurance of the product.

We seek to exploit techniques that extract discriminatory information [8] and correlate with the perceived quality of images at a low computational cost. These techniques must be capable of discriminating images which are almost similar in terms of surface appearance. In this case texture is viewed as key to the successful characterisation of the surface appearance of paper. The HVS uses a pixel and its local context, i.e. it compares pixels in a local area and it also notes differences in areas separated by some distance in an image when assessing the surface of an object. This neighbourhood attribute must be incorporated in the proposed vision system. The reason for using texture is that the HVS exploits this phenomenon to discriminate different objects [9]. Humans discriminate between objects apparently almost effortlessly. However, the automatic description (by a computer vision system) of these patterns is complex.

The thesis will study a system that aims at capturing what the human operator sees when classifying paper in terms of surface appearance. It will attempt to strike a balance between computational cost and high discrimination ability.

This thesis does not address the effect of drift due to maintenance issues on the quality of paper.

This thesis does not discuss optical techniques despite their high speed because they suffer from scatter. Additionally, this work is not focusing on the specific odd features caused by a mark on the paper, or large defects like creases on the paper due to handling. Small regular features of the size of 1 mm up to the 1.5 mm size are interesting. The focus is on the general look of the paper. The manufacturer's criteria in failing paper is the uneven distribution of grains on the surface of paper.

Since the fibre is projected onto the wire mesh, there is direction on the surface of paper. The *cross direction* (CD) is periodic hence the Fourier transform might be suitable for characterising it. Blobs show a random distribution on the surface of materials and thus they tend to perform well in the *machine direction* (MD).

1.2 Related work on the assessment of paper and related material

In this section related work based on computer image analysis and that based on light transmission methods will be discussed.

1.2.1 Related work based on computer image analysis techniques

The early paper quality assessment techniques were based on the light transmission method introduced by Davis in 1935 [10]. These techniques were used for measuring the formation of paper defined as a variation in the mass and thickness of a paper. Formation is a poor indicator of the surface quality of paper. This section also includes few selected past works on formation.

Weszka et al [11] characterised the surface quality of materials using features from the co-occurrence matrices that include entropy, standard deviation of entropy and maximum correlation and they were 0.937, 0.927 and 0.913 respectively correlated with the judged quality. The grey level difference method's (GLDM) entropy achieved more than 82% correct classification. However, they do not mention the type of material used and thus it is difficult to compare their work with ours. Conners et al [12] used first order statistics that include the mean, variance, skewness and kurtosis of the grey levels and a measure based on the SGLDM in combination for an automated lumber processing system that identifies and locates surface defects in wood and 88.3% correct classification was achieved. An Upstream Human Inspection (UHI) station which relies on skilled manpower was used to enhance indistinct defects with a black felt tipped

pen and non-flaws were suppressed with a reflective marking paint. In terms of paper surface appearance characterisation for quality, this technique is not suitable because there is too much human interference and therefore its results are subjective. Moreover, the focus of this thesis is not on specific defects, but on the general look of the surface of paper.

Siew et al [13] when assessing the wool carpet wear used the SGLDM's energy, entropy, Inertia, Local homogeneity and correlation; the GLDM's (grey level difference method)) contrast, angular second moment, entropy, mean and inverse different moment and the gray level run length statistics (GLRLM): long run emphasis, short run emphasis, run length non uniformity, gray level non uniformity and run percentage.

Hon-Son et al [14] used the SGLDM contrast feature to categorise metal surfaces into classes of different roughness using the tree classifier which resulted in 90% correct classification. Metal surfaces reflect more light and thus contrast is a useful measure. However, this feature alone is not sufficient to characterise the "ridge/valley" on the surface of paper and hence more measures are needed in order to capture the information.

Lobo et al [15] successfully discriminated four forest types using the SGLDM's angular second moment, contrast and entropy computed using a 3×3 pixel window. However, these four classes in images were visible and thus it was expected that these features would solve the problem. In the context of paper, the visual difference between the samples in adjacent classes is very small hence determining the features that would separate them is a challenge as would be seen in chapter 7.

Thierry et al [16] used the SGLDM energy and entropy on hardwood and softwood hand sheets. The computation was based on the transition of basis weight variation instead of the grey level transition. An increase in entropy resulted from an increase in the grey level transitions.

Chen et al [17] used the Gauss-Markov random fields (GMRF) to model the texture of a textile fabric. The implementation involved partitioning the image of the fabric into non-overlapping neighbourhoods where each neighbourhood was classified as defective and non-defective based on a likelihood ratio test or the maximum likelihood estimates (MLE). Thus the result was postprocessed using the statistical hypothesis testing on statistics derived from the model. In terms of classifying paper, this method is not suitable as the computation of the MLE of model parameters in each neighbourhood is computationally expensive.

L. Macaire et al [18] presented a system that automatically uses grey level extrema to analyse the local uniformity of a strip coating on a continuous galvanising process. The system captures data, processes it and analyses it whilst the strip is moving under a camera. The number of grey level extrema is then counted. In terms of paper classification, a better approach, the specific perimeter method which also relies on the counts of grey level values that are above a certain threshold has been adopted for this thesis.

Conners et al [8] proposed an automated visual inspection system which uses a scanner to collect visual data as the specimen passes through. This data feeds into the processor which in turn directs the sorter to either accept or reject the specimen if it is "good" or "poor" respectively. Paper needs to lose moisture before it can be characterised thus an online system is not suitable.

1.2.2 Techniques based on light transmission

This section presents the paper quality measurement techniques based on light transmission, commonly known as "look through" methods.

The implementation of the "look through" method involves directing a beam of light through an aperture onto a sample and the transmitted light is collected by a photo cell whose outputs are the direct current (DC) and the alternating current (AC). The former current is proportional to the average transmittance of the sample whereas the latter is proportional to the variations in mass per unit area of the paper. Formation in the context of paper is the ratio of the AC to the DC component. The comments regarding the "look-through" method with respect to the task domain are dealt with at the end of this section.

The first known "look-through method" introduced by Davis et al [10] scans a sheet of paper along a line using a light spot of about 2mm in diameter and in the process detecting variations in the intensity of transmitted light [19]. An extension to Davis' work is the Thwing-Albert and the Quebec North Shore-Mead(QNSM) [20] testers among others. The other formation testers include the PIRA and Kallmes's Micro-Formation tester [21].

Kallmes' micro-formation tester [21] measures the point to point mass distribution of a paper. Its photo detector output is divided into 64 optically determined basis weight classes each differing from the next by 1% of the grey scale. Micro-formation in this context is defined as the ratio of the peak height to the base width. Its disadvantage is that it is a 1D technique yet the surface of paper is 2-D. The Quebec North Shore-Mead (QNSM) formation tester evaluates

the power spectra and the distribution of spacings between floc centres and also measures wire mark pattern.

Levitin et al used a "look through" method that traces a curve by a spot of light 10mm in diameter. The output curve corresponds to the varying optical density and it is used for discriminating samples based on formation. The limiting factor is the bigger spot size which misses finer detail and also the waving characteristics of light which introduces noise. Furthermore, the result must be analysed by the human element and thus it fails to replace the human element with a machine.

Corte et al [21] presented a "look-through" method that determines the variance of the basis weight. Their claim is that 75% of basis weight variance is due to manufacturing and the remainder is due to the randomness of the process.

The density of summits(DOS) [22] defined as the average of summits in a unit area of the surface is viewed as a precursor to the specific perimeter method (SPM) covered in chapter 5. The analogue to the SPM perimeter and field of view (FOV) is the DOS' contour length and area respectively. Their assumption was that the paper surface is random and isotropic. The surface appearance of the paper being examined in this thesis has a random surface appearance and thus there are striking similarities with our approach. Statistics of the distribution of summit heights, contour length and excursion area are used to compute the percentage for the void.

The beta rays [23] as a light source depends on the amount of mass to be traversed and they work even in colour and where there is density variations in paper. Beta rays were replaced by white light as a source of light in paper inspection during the $1970s$. The reasons for their replacement were their spot size which is larger than that of white light and in addition beta rays are expensive. Demeyer [23, 24]used beta radiography to measure formation. However, beta radiography is affected by the film-type, exposure, developing time, temperature and developer concentration all of which impact badly on the paper quality inspection result.

In summary, the "look-through" techniques whilst they are intuitively appealing, they are prone to human error in interpretation. The other limitations are the spot size, the scatter and oscillation characteristics of light which contribute to classification error. These "look through" methods measure formation and not the surface appearance and thus they are not suitable for the assessment of paper surface appearance. Furthermore, they rely on the viewer's skill and experience and their result is thus subjective. Assessing the internal structure, i.e. floc distribution

is not the goal of this thesis.

1.2.3 Thesis Structure

The previous sections in chapter 1 presented the aims of the thesis which include developing an automatic classification system for assessing the surface appearance quality. Related work that has been done in the past using light transmission techniques and a few using computer image analysis has been discussed. There has been little from the computer analysis because not much effort has been done on the surface appearance quality, but formation.

Chapter 2 gives background information on image processing with more emphasis put on filters. The last part of this chapter is a general discussion on features, feature extraction, texture analysis and a brief discussion on image capture.

Chapter 3 presents background theory on spatial feature extraction techniques.

Chapter 4 presents spectral feature extraction techniques that include multiresolution techniques and linear transform methods.

Chapter 5 gives a brief overview of fractal dimension and lacunarity. The back ground theory on the specific perimeter method (SPM) is also given. In addition, an experiment for the optimisation of the size parameters for the SPM is also presented in this chapter.

Chapter 6 is on feature selection. The background theory on classification with the focus on linear and nonlinear neural networks is also covered.

Chapter 7 presents a discussion on classification and strategies used for interpreting the results. Extensive experimental work carried out on paper images is also included in this chapter.

In chapter 8 a discussion on the results and contribution to knowledge is presented and the conclusions are drawn. Further work is also proposed.

The schematic in Figure 1.1 captures the structure of the thesis.

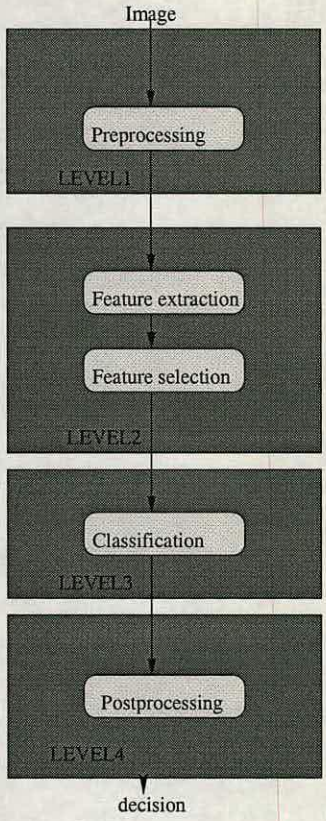


Figure 1.1: *This flow diagram illustrates the important parts of the vision system that forms the subject of this thesis. The input are images and the output is a decision.*

1.3 Thesis and Contribution

The novelty in this thesis in Fig 1.1 will be in 2 up to level 4. The combining of multiple features in some way should form a major part of this thesis. An extensive discussion on the contribution of this thesis is given in the thesis conclusion in chapter 8.

This thesis examines the suggestion that an automated paper-classification system based upon multiple texture-based features and trained to emulate a human operator can approach the performance of the human visual system.

Chapter 2
Image Pre-processing, Texture Analysis and Feature Extraction

2.1 Chapter Introduction

This Chapter is on image capture, image processing, texture analysis and feature extraction. The aim for image capture and image processing is to get a high quality image. In practice there are no perfect images, but there are only acceptable levels of imperfections. This chapter includes a comparative study of different image processing techniques in terms of their usefulness in removing noise in images of manufactured paper. The noise removal techniques (filters) are many and varied and thus the focus will be on techniques that effectively deal with gradient illumination and high frequency noise.

Images contain a lot of information in the form of pixels and measurements taken from image called features can reduce this information and result in a less computationally expensive vision system. The adequacy of texture with respect to information it has about the image and consequently, its usefulness to classification is also discussed.

A theoretical treatment of texture analysis presented in this chapter lays ground for feature extraction techniques discussed later in chapters 3, 4 and 5. The rationale for determining these preprocessing techniques is the need to eliminate gradient illumination and other low frequency noise. This is because natural images have Gaussian or additive noise and this is also true with manufactured paper. There is also likelihood of there being high frequency noise from external sources like, a speck of dust or stray particles falling on paper.

2.2 Image capture

An experimental procedure that results in the capture of high quality paper images is useful for building a vision system. "Good" throughout this thesis is used to mean an image captured from a sample that is classified as good. Similarly, a "poor" image is an image captured from

a sample that has been classified by the human expert as poor. There can be a powerful image analysis tool but if it is used on an image that has not been properly captured, all the work done is futile. In critical areas like medical imaging the need for a high quality image is critical.

The bench mark is observing on captured images features seen on the samples when they are viewed under angular illumination. The detailed experimental procedure on image capture is included in a separate section in chapter 7.

2.2.1 Background on Image capture

An image is a matrix that consists of discrete rectangular cells called pixels, an acronym for "picture element". A pixel's brightness value is called a grey level value. A grey-scale image is one where the only colour is the shades of grey which decrease progressively from 0 to 256 (from grey to white). For an 8 bit image the range of values of a grey-scale is from 0 to 255. The pixel can take any value within this range.

The pixel in a digital image loses the grey level variation that characterises its surface in a continuous image. Capturing images using high resolution digital cameras minimises such losses. This camera must have square pixels and it must also be capable of working under low light intensity requirements (working indoors). Whereas hexagonal pixels [25] have better connectivity, the technology widely available (CCD and scanners) supports rectangular pixels.

An ordinary camera needs a scanner and a slide for digitising the image to the resolution of interest. The digital camera still lags behind the ordinary cameras in many spheres. However, because the former simultaneously captures and digitises the images, it is viewed as the quicker and convenient tool in machine vision.

Illumination plays an important part in the capture of images. However, techniques that depend on the direction of the texture are affected by changes in the angle of lighting. Directional illumination during image capture acts as a directional filter of texture [26]. Thus both the profile of a surface and the illumination angle are critical when working on texture. Changes in illumination direction occur when the light is very close to the sample. Thus during image capture the distance of the light to the sample will be optimised.

The potential sources of light used for illuminating the paper include X-rays, infra-red, ultra-violet and white light among others. X-rays, infra-red and ultra-violet are expensive sources of

light and are only used when white light fails.

The field of view (FOV) in machine vision is a parallel of the fovea in a human eye. The image in the visual periphery of the eye is indistinct. The saccade (a reaction that moves the eyeball) of the eye puts an image in the fovea for closer scrutiny. This process is called fovealisation [27]. In addition, the distance of the eye from the paper sample when one of the two objects being viewed comes out of the line of sight is called the visual angle. The FOV of the camera was chosen along this line of thinking after an exhaustive optimisation procedure. Thus the spatial resolution of the human visual system which also decreases with an increase in the distance of gaze might be emulated.

This optimisation was an attempt to match machine operation to the human visual system. The detail on the image capture experimentation is included in chapter 7.

2.2.1.1 Preprocessing

This section presents image processing techniques with respect to their application on paper images. The focus is on noise removal. The elimination of many noise sources during image capture can reduce the amount of preprocessing needed.

2.2.2 Introduction for Preprocessing

Real world signals are probabilistic and thus desired data tends to be obscured. The likely sources of noise are variations in the camera sensor sensitivity (in this example the charge coupled devices (CCD)), fixed pattern errors in the CCDs and noise due to environmental variations.

Uneven illumination during image capture is also a source of degradation (noise).

Image independent noise is described by an additive noise model of a random Gaussian distributed noise $n(x, y)$ in Figure 2.1 and this is also typical of the detector noise. This noise is randomly distributed over the frequency domain.

The camera-captured pixel value is given by the following relation:

I(x,y) is the sample or raw signal, D(x,y) is a filter used for preprocessing the image , $f^*(x, y)$ is an approximation to the true pixel value f(x,y). C(x,y) is the noise introduced by the camera.

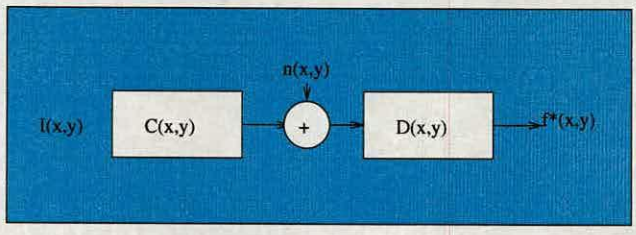
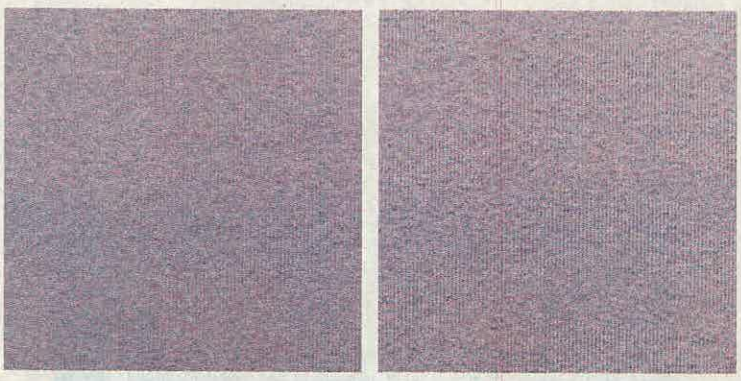


Figure 2.1: *This is an additive noise model of a camera captured image and a filter that estimates the original intensity.*

The information on the feature size of interest influenced the choice of the filter size to be used. The features on the paper that are above the size of interest in the context of paper must be eliminated as they are due to causes other than the process that produces paper. Filters must be optimised such that they remove just enough noise otherwise a significant portion of the signal might be removed as well. Thus filtering is key in image processing. The images Figure 2.2(a) and Figure 2.2(b) are original images captured from manufactured paper. The detail on image capture and preprocessing is included in chapter 7.



(a) **(b)**

Figure 2.2: *The images Figure 2.2(a) and Figure 2.2(b) are original good quality and poor quality images respectively.*

2.2.3 Wrapping and Saturation

The maximum pixel value in an image is limited by the format of the image. The output of most spectral techniques exceed the dynamic range (*255 for an 8-bit image*) of the image. This pixel

overflow is solved by *wrapping around*. Wrapping around involves subtracting the maximum grey-scale value range from the overflowing pixel value. A key advantage of wrapping around is that it retains the differences between the pixel values. However, its weakness is in that a pixel value passing through the maximum value might assume a minimum value.

Alternatively, if only a few pixel values exceed the maximum value, the overflowing pixels can be set to a maximum value. This effect is known as *saturation*. However, if all pixels exceed the allowed maximum value, the result is an image of constant pixel value hence a loss of information occurs. In summary, guarding against wraparound and saturation is a very useful step in image processing.

2.2.4 Filters

This section presents background theory on filters for preprocessing images. These filters include spatial filters and convolution filters. The output of these filters is a linear combination of grey levels in a local neighbourhood of the current pixel inclusive of the current pixel. The current pixel gets replaced by this new value. The neighbourhood is chosen to have an odd number of row and column pixels and thus the current pixel becomes the central pixel. A larger neighbourhood can be computationally expensive.

Another approach called block-by-block processing divides an image into sub-images and then computes a statistic for each subimage. The output from each subimage region is later combined with others. Processing is only determined once for the whole subimage, consequently, it is computationally cheaper than the area process (filters based on convolution). It also reduces the noise without significantly blurring the image. However, it suffers from the blocking effect.

In filtering, redundancy present in images can be exploited by replacing an isolated pixel by a median value [28] of pixels in the neighbourhood. This is a non-linear filtering process.

Smoothing and gradient operators are commonly used image preprocessing approaches. The former is an analogue of high frequency suppression in the Fourier transform domain as it suppresses noise and also desired information in the form of edges. The latter's analogue in the Fourier domain is the suppression of low frequency. It is based on local derivatives and produces a large magnitude at abrupt changes on the surface of an image, consequently they enhance high frequency information and also noise. The low frequency components correspond to a homogeneous surface and background whereas the high-frequency components correspond

to edges and small details in the image. Thus a challenge is coming up with a filter that preserves information whilst keeping noise to a minimum.

2.2.5 Low frequency suppression techniques

In this section techniques that enhance the high frequency detail in the image are presented. Low frequencies tend to obscure the true pixel values of paper images.

2.2.6 Introduction

High frequency carries both the desired information and noise. The techniques that enhance detail [29] like the high pass filter do enhance noise as well. What is critical are techniques that isolate the high frequency information from the low frequency information. In paper images, the low frequencies are the illumination gradient and the overall contrast and they must be attenuated. The results of high pass filtering paper are shown in Figure 7.4(b). The bench mark for noise removal was the disappearance of the illumination gradient on the surface of paper. A high pass filter derived from frequency filters [30] is ideal because the frequency of interest can be set precisely as a cut off frequency. However, an equivalent spatial filter is preferred to the frequency filter in applications that do not need high frequency precision. This is because implementation of the spatial filter is not complex although it might be computationally expensive when larger masks are used. In the spatial domain results are always better. However, since the features of interest are small, thus a smaller spatial filter is recommended for use in this thesis. The preference of the spatial filter over the frequency filter is from the Fourier theorem which says that multiplication in the frequency domain is equivalent to convolution in the spatial domain [30].

This relationship between spatial and frequency domain filtering is established by the convolution theorem.

$$f(x,y) * h(x,y) = F(u,v)H(u,v).$$

where $f(x,y)$ is an image and $h(x,y)$ is an impulse response.

The LHS is obtained by taking an inverse Fourier Transform of the RHS. The RHS is obtained by taking a forward Fourier Transform. Thus convolution in the frequency domain reduces to multiplication in the spatial domain and vice-versa.

The frequency domain carries a significant degree of intuitiveness regarding the specification of filter parameters. It also depends on the size of the spatial filter mask and this is usually answered with comparable implementations between the spatial and the frequency filters.

If both filters are of the same size, it is more efficient computationally to do the filtering in the frequency domain. However, filtering in the spatial domain using small masks can be useful too. As an example, filtering based on a Gaussian is useful because its shape is easy to specify and the forward and the inverse FT of the Gaussian are real Gaussian functions. This is illustrated by the following equation.

The result of the forward FT: $H(u) = A^{-u^2/2\sigma}$; The result of the inverse FT: $(x) = \sqrt{2\pi}\sigma A e^{-2\pi^2\sigma^2 x^2}$; It can be seen that there is an inverse relationship between the functions. Ideal filters could be used but they produce ringing (filter showing multiple peaks in the spatial domain) in the spatial domain. A Gaussian does not suffer from ringing, however, because it is a continuous function, it might not be suitable for use on digital images. An approximation of the Gaussian, the Butter-worth filter [30] is robust to ringing and it is thus recommended.

2.2.7 2-D Gaussian Smoothing

The Gaussian filter [28] is a bell-shape low pass filter built from a Gaussian function. This filter removes both detail and noise and the result is a blurred image. The Gaussian distribution of the data is assumed to have a mean of zero. The application involves convolving the image with an optimal Gaussian kernel. Smoothing increases with an increase in the standard deviation σ of the kernel in (2.1). The relation for the Gaussian filter is given by:

$$\mathrm{G(x,y)} = \frac{1}{\sqrt{2\pi}\sigma} e^{-(\frac{x^2+y^2}{2\sigma^2})} \tag{2.1}$$

where x and y are the image coordinates. The output of each pixel in the neighbourhood is weighted towards the value of the central pixel. Gaussian smoothing therefore preserves edges better than a mean filter of the same size. The key is that a Gaussian has a good frequency response as it does not incur ringing in the spatial domain. Moreover, frequencies in the filtered image can be known precisely. A Gaussian in practice is zero more than 3σ [25] (instead of being zero everywhere) as pixels further than this have negligible influence. This is a minor

weakness of this filter. The application of a Gaussian function is in the Gabor filter [31], in the Laplacian of a Gaussian(LOG) and in blurring images.

The visual cortex in the brain has approximately a Gaussian response [32] and thus a Gaussian filter is biologically plausible. A Gaussian filter fails in the presence of impulsive noise although this might not be an issue in the case of paper. In addition, this filter can be computationally complex when a larger σ (size of kernel) is used. Overall, this filter is potentially useful for filtering paper images.

2.2.7.1 Laplacian of a Gaussian

The following is an equation for a Laplacian of a Gaussian:

$$\text{LOG(x,y)} = \left(\frac{x^2 + y^2 - 2\sigma^2}{\sigma^4}\right) e^{\frac{-(x^2+y^2)}{2\sigma^2}} \tag{2.2}$$

x and y are pixel co-ordinates in an image and σ is the filter's standard deviation. The Laplacian of a Gaussian [25] is a filter that enhances the high frequency information of an image (edges, lines on an image) and it is thus a high pass filter. Its implementation involves the convolution of an image with a Gaussian function [28]. The Gaussian filter's standard deviation σ determines the amount of smoothing. The next stage involves convolving the Gaussian-smoothed image with a Laplacian kernel to produce the gradient of an image [28]. The Laplace transform could not be used alone because it is a gradient filter which makes it sensitive to noise.

The Gaussian was used prior to the use of the Laplacian to suppress noise. However, the LOG is sensitive to noise as it amplifies high frequencies. Pre-convolving the Gaussian with a Laplacian filter to form a hybrid filter is economic and computationally cheaper as there is only one convolution with an image. A large positive value is normally added to a LOG filter's response to eliminate negative values resulting from the filtering. As σ increases filtering becomes insignificant and the LOG progressively approaches the Laplacian kernel. In the context of paper, σ is a smoothing parameter and different σ expose different scales of features. This is key in computer vision as the information of interest might be easily captured from one scale than the other. An optimisation of σ and also the Gaussian window size are therefore critical for the success of filtering. However, not all the noise is suppressed by the Gaussian, and, a Laplacian

being a high pass filter, also enhances noise. Furthermore, the use of the LOG filter produces ringing on the output image. In summary, the LOG filter is not ideal for this problem

2.2.7.2 The Unsharp masking filter

The unsharp masking filter enhances high frequency detail and it is thus a high pass filter. The implementation of the unsharp filter [25, 33] is in three stages, the first stage involves the smoothing of an image and the second involves subtracting the smoothed image from the original image. The final stage involves adding a proportion from the original image to the image obtained in the second stage (the gradient image). The final output is an image with enhanced detail.

It is intuitively appealing, but the three stage implementation makes it computationally expensive especially as we had a database of 1685 images. Furthermore, the fraction of the original image added to the difference image is obtained through guess work. This might compromise the result. In addition, an image produced by this filter exhibits ringing at high contrast edges.

This filter finds application in the printing industry [33].

In summary, an unsharp filter can attenuate the low frequency signals whilst allowing the high frequency information to pass untouched. The three stage implementation makes it computationally expensive. Furthermore, many stages of implementation can lead to loss of information (low pass filtering, subtracting etc).

2.2.7.3 Frequency filter

A frequency filter selects frequencies either that are below or above a certain frequency (threshold) called a cut-off frequency. A Butter-worth filter is one commonly used frequency filter. There is a high pass filter, low pass filter and a bandpass filter version of a Butter-worth filter. The Butter-worth has the advantage of having the flattest pass band and also having a response that does not exhibit ringing in the spatial domain. The implementation of the Butter-worth filter involves getting a Fourier Transform of an image and then multiplying the result with the filter function. The resulting image is re-transformed into the spatial domain. The equation for frequency filter is given by:

$$g(x,y) = f(x,y)H(x,y). \tag{2.3}$$

H(x,y) is the filter function, g(x,y) is the filtered image and f(x,y) is the input image. The frequency filter can be approximated by the spatial domain filter [30] and the former is only used if no proper spatial domain kernel can be found and also when the focus is on a specific frequency in the image.

2.2.8 The High Frequency noise suppression techniques

In this section techniques that remove high frequency noise are discussed. Noise normally resides in high frequencies in an image. The high frequency components on an image also include edges which are part of the desired information and they show up as white patches on the image. The aim is to suppress the noise components without the loss of true data. The filters in this section were judged on the ability of preserving true data whilst attenuating noise.

A low pass filter is a zero-phase-shift filter [30] and only slight low pass filtering must be applied for high-frequency image regions otherwise the image will be blurred. However, an insignificant amount of the signal is attenuated by low pass filtering as a larger portion of information resides in the low frequency regions owing to the assumption that neighbouring pixels are highly correlated. The reason being that there is a gradual change from one point to the other in many natural surfaces, consequently neighbouring grey levels have almost equal grey level values.

The ideal low pass filter suffers from ringing (filter showing multiple peaks in the spatial domain) and this shows up along the intensity edges of a filtered image. Increasing the cut-off frequency results in a greater number of frequency components and hence more detail, with less blur and less severe ringing but with more noise.

The low pass filter version of a Butter-worth filter [30] has the same attributes as those mentioned earlier for the high pass filter version. The Butter-worth filter is computationally cheaper than a spatial filter. The Butter-worth filter is also independent of the filter function thus it is recommended for wide low pass filtering whilst the latter is recommended for narrow low pass filtering. The Butter-worth has a lowest passband region and thus it has the least attenuation

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Table 2.1: *This a* 3×3 *mean filter*

over the desired frequency range. Its transfer function does not have a sharp discontinuity and that establishes a clear cutoff between the passed and filtered frequencies.

2.2.9 Mean filter

The following is the relation for the mean:

$$\text{mean(x,y)} = \frac{1}{N} \sum_{k=x-1}^{x+1} \sum_{k=y-1}^{y+1} f(x,y) \tag{2.4}$$

N is the total number of pixels in a neighbourhood, f(x,y) is the image, x and y are pixel coordinates in the image.

The mean filter [25] shown in Figure 2.2.9 is a low pass filter used for removing high frequency noise from an image. The implementation of this filter involves choosing a neighbourhood, normally a 3×3 size and each pixel value in an image is replaced with the mean value calculated from its neighbours including itself [28].

The type of noise expected in paper images is Gaussian noise and the mean filter effectively removes most of this noise. However, if a larger mean filter is used, both noise and the desired high frequency detail are removed. In addition, a larger mean filter is computationally expensive. In the presence of impulse noise in a neighbourhood, the mean value obtained is either too small or too large compared to neighbourhood pixel values and it is therefore not representative of the neighbourhood. Additionally, the mean on edges interpolates new values and as a result edges get blurred. This filter suffers from ringing in the spatial domain. The mean filter is therefore not suitable for application on manufactured paper. The images in Figure 7.4(a) were filtered using a mean filter.

An extension of the mean filter involves comparing each pixel to the average of its neighbourhood and if it differs by more than a specified threshold value, it is replaced by the average

value. However, the problem is selecting the threshold which must ensure that true pixels close to the maximum or minimum are retained and that true edges within the image are also retained. However, in the context of paper this does not improve the mean filter's utility.

2.2.10 Suppression of impulsive noise

Impulse noise [28] has extremely high values compared to pixels in its neighbourhood. These noise pixels must be forced to conform to their neighbourhood. An example of a neighbourhood is shown in Figure 2.3. The median filter and the morphological filter among others are techniques that eliminate impulsive noise.

The following Figure 2.3 is an illustration of a neighbourhood.

A	B	C
H	i,j	D
G	F	E

Figure 2.3: *This is a* 3×3 *neighbourhood. The pixel (i,j) is called the central or current pixel and the rest are neighbourhood pixels*

The rectangular neighbourhood is popular although its use can damage edges in an image.

2.2.10.1 Morphological filters

The morphological filter [28] consists of closing and opening and the opening and closing filters. These filters preserve edges and eliminate high frequency noise.

The application of erosion immediately followed by dilation is called opening. The corollary is closing. Erosion in an opening removes isolated pixels as well as boundaries of the objects whilst dilation restores most of the boundary pixels without restoring the noise. The isolated pixels are thus removed. Morphological filters can outperform the median filter in removing this type of noise. However, its optimisation is complex as it uses structuring elements.

2.2.10.2 Median Filtering

This section gives background theory on the median filter and its relevance to preprocessing manufactured paper images. The median filter [28, 30] is a nonlinear filter that removes impulse (a random occurrence of extremely high pixels values) noise, the "salt" and "pepper" noise and random noise from an image. Unlike other low pass filtering approaches, median filtering preserves detail (desired information).

Its implementation involves sliding a window along the image and at each position pixels are sorted in the neighbourhood into ascending order and the middle pixel value is used to replace the current pixel value. An isolated pixel will always be removed because extreme pixels in a sorted neighbourhood are placed either at the bottom or top of the scale. The current pixel is replaced by a pixel within the same neighbourhood hence image detail is preserved. However, the median filter fails where more than half of the pixels in a neighbourhood have been corrupted by noise. Additionally, the median filter is computationally complex due to the sorting of the the pixels within the neighbourhood procedure. Furthermore, features with a frequency with a span less than the filter size are smoothed. However, using a smaller median filter several times instead of one large median filter removes most of the additive noise with less loss of detail and is less computationally expensive and also speeds up computation. Thus a median filter is potentially useful for this work.

2.2.11 Fuzzy filters

This section gives a brief overview on fuzzy filters [34] which are still new in the field. Their implementation involves sorting the pixels in a neighbourhood numerically. A maximum is then computed with respect to an appropriate fuzzy measure (AND or OR). The output and input parameters are trained using the LMS and the gradient descent respectively which prevents the latter from getting trapped in a local minima. The key feature of these filters is linguistic rules (fuzzy IF-THEN) which are combined with the numerical information (input-output pairs). The fuzzy system uses its relation and properties of the sample and the hypotheses from the labels of the neighbouring samples to create a new hypothesis and hence decisions about the classes.

Traditional filters that reduce noise also blur the edges. In contrast, fuzzy filters combine edge-preservation and smoothing [34]. Increasing the size of rule base improves this filter's performance up to a point, beyond which it becomes susceptible to poor performance due to rule base

explosion. As a consequence, training routines fail to adequately explore the large parameter space and the resulting system becomes computationally expensive. The solution to this involves the use of a technique called simulated annealing [34] a training technique that selects an optimal non-exhaustive rule base.

Fuzzy filters generalise the averaging filter, morphological filters and the median filter. In summary, whilst fuzzy filtering can outperform median filtering, its major disadvantage besides the rule base explosion, it needs training and training normally produces suboptimal solutions.

2.2.12 Summary: Filters

The mean filter, the Gaussian smoothing filter and the median filter are commonly used low pass filters. The mean filter fails to remove impulsive noise because the corrupted pixel would be significantly higher than the neighbourhood pixel values. The mean filter is ideal for smoothing an image degraded by additive noise. However, it reduces the magnitude of an intensity gradient. In the case of high pass filters if there is no constraint on frequency, then spatial filter form of the high pass filter is the best choice as it is easy to implement although it can be computationally expensive.

However, the median filter is preferred to the other 2 filters because in addition to removing random noise it also removes impulsive noise and furthermore, this filter re-uses pixel values in the neighbourhood.

2.2.13 Binarisation Techniques

This section presents image binarisation approaches and their application. Binarisation is important in this study because a technique called the specific perimeter method that is covered in chapter 5 is based on binary images.

2.2.13.1 Introduction

2.2.14 Histogram

A histogram [30] is a graph that consists of bins and each bin contains distinct grey levels in an image. In an x-y plot, the y-axis measures the frequency (the total number of pixels in each

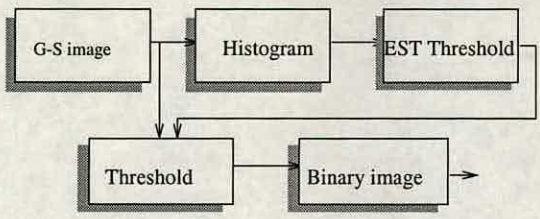


Figure 2.4: *This is an illustrative schematic for the binarisation of an image. "G-S" is the input grey scale image, "EST Threshold" is estimated from the "Histogram". The final "Threshold" once found is used to generates a binary image.*

bin) whilst the x-axis is graduated in the grey level values and for an 8 bit image, there are 256 possible distinct grey levels and hence data is distributed amongst 256 bins. Each bin has an index that corresponds to a distinct pixel value. The process of allocating pixels continues until all pixels in the image have been assigned to corresponding bins.

In an image that has been transformed, pixel values in the new image should not exceed a grey level value of 254 otherwise clipping could occur unnoticed. For example, a pixel value of 256 and above will be recorded as 255 due to saturation or a small value in the case of wrapping. Thus useful information is lost. Thus scaling the resulting image such that the largest value in the new image does not exceed the grey level value of 254 eliminates chances of losing information through wrapping and saturation. A narrow histogram means that data is concentrated in a few bins and hence the surface might be uniform. In contrast, a wide histogram might mean that data is distributed throughout the grey level range of the image and thus the image surface might be non-uniform. The histogram is also useful in assessing the distribution of data for potential class overlap.

The two approaches that are used for modifying a histogram are equalisation [29, 30] and contrast stretching [25] and the assumption is that the image has to use the full pixel range to display maximum contrast. Contrast stretching stretches linearly the pixel values of an image that has a narrow histogram. In an image where the entire frequency range is used, contrast stretching might not be useful.

In contrast, histogram equalisation evenly distributes pixels over the whole intensity range and the resulting image has an increased dynamic range and contrast. Whereas the goal of histogram equalisation is getting a flat histogram, this is only possible when applied on an undiscretised image. Histogram equalisation enhances detail by removing the effects of first order statistics.

The structure of information in the image remains unaltered. Histogram equalisation finds use in an image with a large number of pixels with similar grey levels. Figure 7.5(b) are histogram equalised paper images and the size of the mask used was 5×5 pixels. Histogram equalisation is not suitable for use on paper images as it blurs edges and other sharp detail.

The histogram of the samples in Figure 2.5 show that an appropriately chosen bin width can reveal the small variations in the data. Thus a clear and detailed distribution as shown in Figure 2.5 is obtained. In this case, the paper samples had almost a Gaussian distribution.

In texture classification the form of the distribution of the data can be key in deciding which classification strategies to adopt. The typical histogram for two non-overlapping sample classes has two distinct peaks (a peak for each sample class). However, the main weakness of the histogram is that it is not sensitive to the rearrangement of pixels in an image.

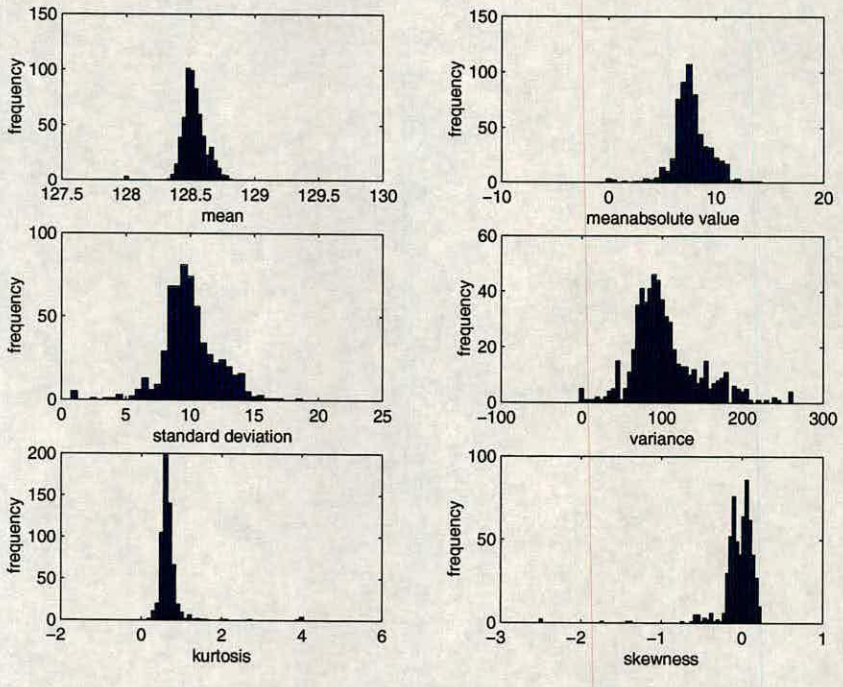


Figure 2.5: *This graph shows a near Gaussian distribution of feature values for the paper samples. The graphs also show the importance of a smaller size of the bin width.*

Binarisation of an image involves selecting a grey level value (GL) such that any pixel in the image with a value lower than GL is assigned a value of 0 otherwise it is assigned a value of 1. This GL is called a threshold value (the lowest point between any two maxima of a data distribution). The binary image must capture desired information from an image. The

motivation for studying binarisation is that a binary image can potentially be used as a classifier. In this case, the most useful binary images are those for the "good" and "poor" images.

This section seeks to find a suitable thresholding mechanism. High thresholds remove both noise and a desired signal whereas low thresholds result in the desired high frequency data and also noise on an image. Thus the threshold must be optimised such that there is a compromise between noise removal and retention of the desired information.

The global thresholding [30] which does not adapt to surface local variations but applies a single threshold to the entire image is not useful for paper images as desired information due to local changes in the image is lost. In addition, it can perform badly where an image was captured under non-uniform lighting [35].

The grey scale and not binary images preserve information. In the former there is no information lost as all processing involves the use of all the pixels. In contrast, the latter is a result of thresholding. For example, global thresholding when used on images that vary locally, it could result in some of this information being lost. Even when local thresholding is used, setting the mask size is not easy and it can never be perfect.

2.2.14.1 Local thresholding

This section looks at local thresholding techniques that include the adaptive thresholding method [28], the Chow and Kanenko [36] method and the iterative selection thresholding [25] technique.

The implementation of the adaptive thresholding approach [28] involves choosing a kernel size and then moving this kernel a point at a time on the image and computing a local statistic at each point, for example a median. The central pixel in each neighbourhood is replaced by 1 if its value is greater than the local statistic otherwise it is replaced by 0. What makes this approach adaptive is the threshold value for each pixel which is dependent on both the current and the neighbourhood pixel values [37].

In the case where the local statistic is the mean grey level value, it excels where the neighbourhood has different grey level values otherwise the mean grey level value will coincide with the current pixel value. The threshold setting level is difficult as the nature of local variations is not known a priori. That is, for the technique to be fully adaptive, the mask size (neighbourhood)

must vary according to the nature of the surface of the image region. Where there is a lot of small variations, then a smaller neighbourhood is ideal. Where there is large variation, then a bigger window (neighbourhood) is ideal. The window size is therefore empirically obtained. This thresholding method is suited to data whose distribution has indistinct peaks.

For data with multimodal distribution, multilevel thresholding [38] which involves dividing the grey scale into bands and then thresholding within each band to determine regions of interest is recommended. The number and the location of the thresholds are determined by trial and error. Thus when the corresponding histogram mode is large the adaptive Chow and Kanenko and not the multilevel thresholding is recommended.

The implementation of the adaptive Chow and Kanenko [36] thresholding involves dividing an image into overlapping subimages and then computing a threshold for each subimage by investigating its histogram.

The assumption is that smaller subimages are more likely to have approximately uniform illumination, thus being more suitable for thresholding.

The threshold for each single pixel in subimages that do not have bimodal histograms is obtained by interpolation. Thus the selection of a threshold per pixel is only ideal where computation time is not critical.

An alternative is the iterative selection thresholding [25] which employs the mean-grey level as an initial guess of the threshold. The mean for all pixels below and above the threshold is computed as Tb and To respectively and their average for each pixel class is the new estimate. The process is repeated and only stops when no change in the threshold is found. This method is computationally expensive and besides, the mean grey-level performs poorly in the presence of an extremely high or extremely low pixel value in the image.

Since we do not know how the surface profile of paper is varying, it is safe to adopt a strategy that tracks such changes. Applying global thresholding, important information will be missed if the surface being characterised varies locally. The choice of the size of the neighbourhood depends on the perceived variation of the surface and the resolution of the image. The higher the resolution of the camera used to capture the image, the larger (in terms of pixel numbers) will be the neighbourhood. This is to facilitate a wider coverage of the space that have different pixel values.

2.2.15 Summary: Binarisation

The adaptive thresholding can capture desired information provided the right mask size, say that of a median filter is used. Methods have been proposed by Weszka et al in [39] for obtaining thresholds in multi modal histograms, [40],[37] and Sahoo [41]. Kohler [42] suggested a thresholding method using intensity contrast between adjacent pixels. Wang [43] proposed a threshold selection based on the histogram of the edge image. A large neighbourhood in addition to being computationally expensive can violate the assumption of a uniform local region (neighbourhood having grey levels with almost the same value) hence poor results might ensue. This is because the larger the neighbourhood, the bigger the distance between pixels at the opposite ends of a neighbourhood. Their grey level values will tend to have a larger difference. The key to the success of local thresholding is the range of grey level values of each neighbourhood and the population of pixels in the resulting binary image.

The recommended local thresholding method in this work is the median value. Local thresholding is useful as it can adapt to the varying pattern on the image. The procedure on choosing the neighbourhood size will be presented in chapter 7.

2.3 The definition of Texture

This section presents texture, a phenomenon that has a contextual property. The motivation to study texture is covered later in this chapter on the section entitled "Why use texture ?". Texture has no precise definition due to its variability [25, 44]. As an example, clouds, an image of grass, a surface made from pebbles, a surface of a wooden table and a surface of sand are illustrations of how varied texture can be. The 2-D image texture shall hereafter be called texture. Texture is a homogeneous visual pattern of a surface of a material which gives it its characteristic appearance. The definition will become clear after its primitives have been introduced in the following paragraph.

The pixel and the texel are constituent elements of texture. Texels, an acronym for "texture elements" are the primitives for the texture of an image. A number of pixels make a texel. Texels are of various sizes, degrees of uniformity and orientation and also have structural or probabilistic arrangement. A pixel is an acronym for "picture element". Pixels in a neighbourhood are arranged either stochastically or according to some placement rule. The ability to discriminate textures is dependent on scale, rotation and changes in illumination used when capturing

the image. Large and small subregions formed by texels characterise coarse and fine texture respectively.

The textural and contextual parameters are used in human interpretation of spectral image information. Context, texture and tone are always present in an image and one property tends to be more dominant than others [45]. Contextual features contain information derived from an image's neighbourhood. Tone is based on the varying shades of grey of pixels in an image. The textures in natural images are often not uniform due to changes in orientation and scale of its primitives. A surface of uniform intensity has zero texture. Thus texture is only pronounced where there is more than one distinct pixel in a neighbourhood. Texture therefore is the spatial variation of grey level values (intensity) in a neighbourhood within an image.

Texture can fully describe the surface composition of an image and the tonal primitives can be exploited as well as spatial interrelationships between them [44]. In addition, when an image has a small or a large variation in tone, the tone or texture become dominant respectively [45]. Consequently, texture-tone specifies both the local properties from tonal primitives and the organisation among tonal primitives. This is a justification for using more features than one to characterise texture. Julesz [9] experimentally found that two textures were only discriminable if their second order statistics were different.

In summary, computational complexity that has been a deterrent to the exploitation of texture measures like the SGLDM is no longer a problem as there are now fast computers. It is hard to tell which pixel contributed to the overall surface appearance of an image. Thus assumptions are often made about the uniformity of pixel values in a neighbourhood (local region in an image). However, this assumption does not always hold in images of natural surfaces because there could be a speck on a neighbouring pixel or a dust particle that will result in this pixel having a value that is very different from other pixels in the neighbourhood.

2.3.1 Categories of texture

Texture is divided into micro-texture and macro-texture. The latter also called deterministic or structural is regular whilst the former also called stochastic or statistical is probabilistic. The noise seen on a television screen is an example of how stochastic texture looks like. Texels in structural texture are arranged according to some placement rules (texel is repeated at predicted intervals). An example of the structural approach is a brick wall. The brick is the primitive

and the pattern on the wall due to bricks that make it is the structural pattern. Natural textures are a mixture of stochastic and structural texture. However, both the structural and statistical methods suffer from scale change.

Texture can also be categorised in terms of strength (strong, weak and constant) [26]. Texture strength is key in the description of textured surfaces. A strong texture has visible and well defined texels. In addition, the spatial interactions between these texels are regular and can be described by the local frequency of occurrence of texel pairs. Weak texture has small spatial interactions between texels. Most textures belong in the weak texture. The constant texture of an image region is where a set of local properties in that region is constant or slowly changing.

Large texels can be distinguished from each other even with small differences between their intensities. However, distinctions between small texels are only visible when there is a large difference between their intensities. Thus in part, texture strength is correlated with coarseness and contrast.

Texture is further classified into coarse and fine texture. Fine texture is characterised by small texels and a large grey level value difference between the neighbouring texels whilst a coarse texture is characterised by large texels consisting of several pixels. As a consequence, texture in the latter has a high degree of local uniformity of intensity over a large area.

Busy [46] and complex texture are other categories of texture. The former is due to an abrupt change in intensity from one pixel to its neighbour and the magnitude of these changes is a function of the dynamic range of the grey scale. If the spatial frequency of changes in intensity is very low, a high degree of local uniformity may be perceived even if the magnitude of the changes is large. The removal of contrast from the spatial frequency information of the image may expose the degree of busyness of a texture. The complex texture has high visual information content and is characterised by many texels that have different average intensities. An example is that of texture composed of sharp edges. There is some correlation between complexity and busyness and also contrast. The key is to build these differences in texture into a classifier.

2.3.2 Why use texture?

In this section the importance of texture in general and in this work in particular is studied. Most surfaces exhibit texture [45] but each surface has unique texture and this is a motivation

for the study of texture for discriminating paper images. As a consequence, texture must assist in achieving high texture classification performance(assignment of samples to one of possible categories).

Texture is the characteristic feature of natural surfaces that is exploited by humans when assessing visual information. Humans use the statistical moments of the distribution of grey level values. The colour seen by humans at a pixel is determined by the current pixel and pixels in the neighbourhood. Foveated imaging must exploit the spatial resolution of the human visual system which decreases dramatically away from the point of gaze (inverse square law) [25]. Thus a vision system based on texture must mimic these attributes. The human eye has the ability to discriminate with little effort between different textures within an image. However, humans are prone to fatigue, stress and boredom hence the need to harness a machine. Thus finding out features on the surfaces of materials which enable the human element to discriminate images easily and succinctly is critical. However, the vision systems are yet to provide adequate tools in the area of texture analysis and texture classification to match human perception. This is a gap that needs to be filled and we aim to fill it.

In the study of texture in the previous sections it has been established that important information for discriminating images is contained in the texture of an image. The spatial relationship between the texels gives the image a random structure or mutually independent characteristic. Rearranging the same texels results in a different texture. Haralick et al in [45] found that texture-context information in an image is contained in the average spatial relationship of the grey levels. The spatial dependence of grey-level values therefore contribute to the perception of texture. Thus the SGLDM [45] exploit this neighbourhood property of texture and therefore might be effective in describing texture.

The detectable differences between data is a function of the variability of texels' size, density and orientation and it is a recipe for high classification performance. However, texture analysis is still elusive (no formalism yet) although exciting for assessing the surfaces of materials. Most image analysis techniques deal with image pixels on a single scale with the exception of multiresolution techniques which characterise different scales of textures.

The application of texture has been on carpets [13], hardwood [12], metal [14], in remote sensing [11], brain imaging [47, 48], estimating crop yields [15], in automated inspection of textile fabrics [49], in defect detection [50], in inspection for automatic of strip coating on a

continuous galvanising process [18] and in surface inspection [8].

Over the past three decades a lot of work has been done in computer vision in search for measures that can adequately capture information from textured images and hence in chapters 3, 4 and 5 texture analysis techniques suited to characterising paper are presented.

2.3.3 Texture Summary

Texture is a homogeneous visual pattern of a surface of a material which gives it its characteristic appearance. Texture primitives, the texels are of various sizes, degrees of uniformity and orientation. The spatial relationship between the texels gives the image structural and probabilistic structure or a mixture of these. These features of texels are key to the discrimination of textured surfaces. Texture is only pronounced where there is more than one distinct pixel in a neighbourhood otherwise there is zero texture. The surface of ice is an example of zero texture. The local spatial variations of intensity with some degree of uniformity is key to texture description and classification.

Most surfaces exhibit texture [45] and each surface has a unique texture, consequently, humans exploit this phenomenon to discriminate different textures. Thus a vision system based on texture has a potential of mimicking these attributes.

There is no optimal scale for all textures, consequently, a multiscale approach might be useful. In the context of paper surface the scale of the texture is more tightly defined and thus scale might not be useful.

The important information is contained in the texture (or the the average spatial relationship of the grey levels[45]) of an image. The application of texture has been in brain imaging [47, 48] and in the inspection for automatic strip coating on a continuous galvanising process [18] among others.

The next section gives an in-depth discussion of feature extraction (taking measurements that capture information contained in the image)

2.4 Feature Extraction

The process of measuring certain attributes of a paper surface is called feature extraction. Feature extraction can involve extracting locally information from every pixel of the input image by performing a convolution using square windows that overlap over the entire image. The resulting single value around each pixel represents the contents of the respective neighbourhood and it is called a feature. Larger windows contain more statistical information than smaller windows. More statistical information is key to high neural classification performance.

Feature extraction therefore captures relationships among the pixels that belong to a similar texture in order to distinguish different textures in images. The extracted features form feature vectors. A set of feature vectors is called a feature space. These features include coarseness, contrast, directionality, linearity, regularity and roughness.

An image contains thousands of pixels and when used as inputs to a classifier, they could result in a very slow classifier. In order to exploit information from pixels whilst saving on the classifier computation time, the information in the image is captured in the form of features. A simple example is distinguishing two people by either height or weight or both. In the context of paper images, the problem is slightly complex and thus many statistical features might be needed.

Features for which the interclass variance is greater than the intra-class variance improve classification performance. Thus features whose means for the two classes differ are useful. Independent features are normally effective in separating differently textured images. Increasing the number of features might improve the classification performance to a certain point beyond which the complexity begins to increase due to the increase in the dimension of the feature space.

Feature extraction techniques that include linear transforms, the spatial and spectral techniques among others are discussed in chapters 3, 4 and 5.

2.4.1 Feature Extraction Summary

This review has demonstrated the importance of feature extraction in computer image analysis as it reduces the amount of information that would otherwise have over-stretched the computing resources if raw data were used as input to the classifier. Features as inputs to the classifier are

key in the success of any computer vision system. Techniques that operate on grey scale images and not binary images preserve information.

The discriminative information for characterising the surface appearance of paper is contained in texture because the surface appearance of paper in the machine direction looks random. Thus features generated from differently textured images should have different values if these images are to be well classified. In case of convolution based techniques, the discrimination capability of extracted features depends both on the window size and images of the surfaces being analysed. An example is in segmentation where small windows can locate the boundaries between different textures. In contrast, texture classification requires large windows to obtain sufficient texture content information because it is a statistical procedure. The properly selected features can potentially construct a discriminant function in the feature space.

2.5 Chapter Summary

The capturing of a high quality image and subsequent filtering is useful in machine vision. The field of view (FOV) used in image capture is an attempt to match the fovea in a human eye.

The pixel in a digital image does not have the grey level variation that characterises its surface in a continuous image (it has a uniform surface). Thus high resolution digital images can minimise such losses. Angular illumination covered in chapter 7 can be useful as it enhances the features on the surface of a material. The sources of noise are variations in the camera CCD and the uneven illumination during image capture among other noise sources.

The variability of texels' size, density and orientation is key to discriminating different textured surfaces. The spatial relationship between the texels gives the image a structural and probabilistic arrangement of texels or both. Thus texture is only pronounced where there is more than one distinct pixel in a neighbourhood.

Most surfaces exhibit unique texture [45] and information is contained in the texture of an image. Texture is scale dependent [44], however, in the context of paper surface the scale of the texture is more tightly defined and thus scale might not be useful.

Textures are visual patterns that are characteristic of the surface and the human visual system exploits it when differentiating textured surfaces. The key is the classification of training samples by human experts that we want to emulate. The human visual system is not well under-

stood and the typical human observer does not exist. An objective measure for the assessment of the quality of an image that matches the human assessment of image quality is still being sought. Neural networks can be used to sort out the most relevant texture features.

The mean filter, the Gaussian smoothing filter and the median filter are commonly used low pass filters. The mean filter is ideal for smoothing an image degraded by additive noise but fails on impulsive noise. In contrast, the median filter removes both random and impulsive noise. In terms of spatial versus frequency filter, the latter is recommended if there is a constraint on frequency otherwise the former is normally used as it is easy to implement although it can be computationally expensive. The information on the feature size of interest determines a filter size and this is useful for filtering out big features which might have resulted from handling by the human expert.

If the transformation of an image results in a pixel overflow, then *wrapping around* which involves subtracting the maximum grey-scale value range from the overflowing pixel value can be used as it retains the differences between the values. Alternatively, the pixel value passing the maximum might be set to a maximum value and this is known as *saturation*. If all pixels are overflowing, then a constant pixel value image results hence a loss of information occurs. In summary, guarding against wraparound and saturation is a very useful step in image processing.

Chapter 3
Spatial Techniques

3.1 Introduction

A framework for extracting features using spatial techniques is presented. This is only a theoretical approach. These techniques comprise the spatial dependence grey level method (SGLDM), the grey level run-length method (GLRLM), the grey level difference method (GLDM), the neighbourhood grey level dependence matrix (NGLDM) and the first order statistics (FOS). Textural features chosen from these techniques should potentially build a computer vision system that correlates well with the human judgement of quality of paper. This chapter gives a brief overview of these techniques.

3.1.1 First order statistics

The first order statistics (FOS) can be defined using a histogram. The implementation of a histogram involves getting a count of the number of pixels M(x) with the same grey level value x in an image and then putting them in the same bin. This count is called the occurrence probability P(x) of intensity and for each intensity it is given by

$$P(x) = M(x)/m \tag{3.1}$$

m is the total number of pixels in the image. The number of histogram bins for an n-bit image in this case is 2^n bins. The features from first order statistics that are potential candidates in assessing the surface quality of paper are:

- standard deviation,
- skewness,

- kurtosis,
- energy,
- entropy

The relation for standard deviation for the data is

$$SD = \sum_{x=0}^{L-1}(x - x')^2 \tag{3.2}$$

where x' is the mean for the data, x is the grey level value and L is the maximum grey level. The standard deviation characterises the way grey level values are distributed around the mean of the data.

The relation for kurtosis is

$$kurtosis = \frac{1}{\sigma^4}\sum_{x=0}^{L-1}(x - x')^4 P(x) - 3 \tag{3.3}$$

Kurtosis measures the relative peakedness or flatness of the distribution of the data. Positive kurtosis means that the data distribution is peaky relative to the norm. Negative kurtosis means that data is flat relative to the norm. Skewness characterises the degree of assymetry around the mean of the data. Positive skewness means that the distribution extends more positive value. Negative skewness means that the distribution extends towards the negative value.

The mean is given by:

$$\mu = \sum_{x=0}^{L-1} x P(x) \tag{3.4}$$

$$\text{energy} = \sum_{x=0}^{L-1} P(x)^2 \tag{3.5}$$

Smooth texture is characterised by a high energy value. Energy is a measure of uniformity.

$$\text{entropy} = \sum_{x=0}^{L-1} P(x) log_2(P(x)) \tag{3.6}$$

L is the maximum possible grey level. x is the grey level value of a pixel. The entropy measures the degree of disorder of the texture considered. Low entropy means a uniform surface whereas and high entropy means a nonuniform surface. This is easy to see from (3.5) as the probability P(x) when x approaches 1 it (P(x) approaches zero. Thus entropy is a measure of non-uniformity.

The advantage of the FOS is that where data is separable, it shows data as separate distribution functions. However, the FOS does not consider the inter-pixel relationships in an image and thus it fails to describe the coarseness of the paper surface. As a consequence, FOS are not sensitive to the rearrangement of pixels in an image. These are its main weaknesses.

Haralick et al [45] asserts that one property amongst context, texture and tone in an image tends to be more dominant. Tone is based on shades of grey of pixels in an image. The performance of FOS is high when the tone and not texture is dominant. This is a deduction from chapter 2 where texture was described as the spatial variation of grey level values in a neighbourhood within an image. Thus FOS might not be useful in the characterisation of the paper surface quality.

FOS has been applied [51] in the detection of osteoporosis (a condition in which bones become brittle) and remote sensed crops [15]. Conners et al [12] used FOS features that include the mean, variance, skewness and kurtosis of the grey levels and a measure based on the SGLDM in combination in the identification and locating surface defects in wood and 88.3% correct classification was achieved.

In summary, the only FOS features that might be useful are the standard deviation and those based on standard deviation like kurtosis. Standard deviation was in the past the recommended tool that was used on paper. The only problem approaches was it not incorporate a constraint that ensures that only the size of acceptable features is included in its computation. Its power lies in its measure of the variation of the pixels around the mean of the data.

Entropy is another possible candidate as it measures the randomness or non-uniformity of texture on the surface of paper.

3.2 The Grey Level Run Length Method

The grey level run length is based on adjacent pixels with the same grey level value.

3.2.1 Introduction

The grey level run length method (GLRLM) [52] assesses the uniformity of the grey level values of adjacent pixels in an image. A "run" is a set of linearly adjacent pixels having the same grey level value. The length is the number of pixels within a run.

The results that follow in Figure 3.1 are glrlm matrices extracted from a test image.

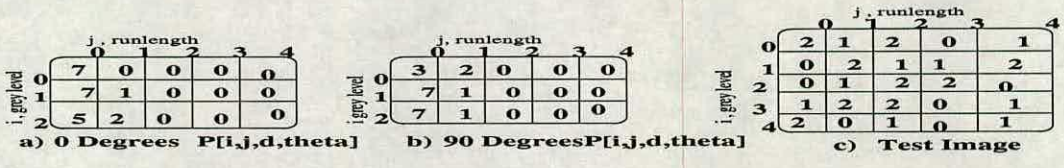


Figure 3.1: *(a) GLRLM distance = 1. (b) GLRLM distance = 1. (c) Test Image.*

The statistics for these matrices have have already been presented in (3.8) to (3.12).

$$R\theta = R(i, j, \theta) \tag{3.7}$$

For a given run direction θ (typically 0°, 45°, 90°, or 135°) the method calculates the number of times an image contains a run of length j ($0 < j \leq J$, where J is the longest possible run in the image) for grey level i ($0 < i \leq I$, i.e. the range of pixel values). The GLRLM thus produces a matrix of dimension $J \times I$. To assess image texture, statistics are computed from this matrix. These comprise of the *long run emphasis* (LRE), *short run emphasis* (SRE), *run length non-uniformity* (RLNU), *run percentage* (RPC), and *grey level run non-uniformity* (GLNU). These may be calculated as follows [52]:-

$$\text{LRE} = \frac{\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} j^2 P_{(i,j)}}{\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} P_{(i,j)}} \tag{3.8}$$

where $P_{(i,j)}$ is the probability of run length of j at grey level i and L is the dimension of the matrix. LRE is sensitive to long runs and a high LRE means coarse texture.

$$\text{SRE} = \frac{\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} P_{(i,j)}}{\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} j^2 P_{(i,j)}} \tag{3.9}$$

(3.9) is a reciprocal of (3.8). SRE is sensitive to short runs. A high SRE means fine texture.

$$\text{RLNU} = \frac{\sum_{j=0}^{L-1} [\sum_{i=0}^{L-1} P_{(i,j)}]^2}{\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} P_{(i,j)}} \tag{3.10}$$

This is the nonuniformity in the runlength axis.

$$\text{GLNU} = \frac{\sum_{i=0}^{L-1} [\sum_{j=0}^{L-1} P_{(i,j)}]^2}{\sum_{j=0}^{L-1} \sum_{j=0}^{L-1} P_{(i,j)}} \tag{3.11}$$

This is the nonuniformity in grey level axis.

$$\text{RPC} = \frac{\sum_{j=0}^{L-1} \sum_{i=0}^{L-1} P_{(i,j)}^2}{N^2} \tag{3.12}$$

N^2 is the number of pixels in the GLRLM image. This number (N) is largest when the runs are all short. Thus RPC is small for fine texture and large for coarse texture. Run percentage is the ratio of the total number of runs to the number of pixels in the image. In a course or fine texture long runs and short runs dominate respectively. The advantage of the GLRLM features is that the length of the runs reflect the size of the texels.

The new GLRLM features [53–55] are:

- high grey-level run emphasis (HGRE)

- low grey-level run emphasis (LGRE)
- short run low grey level emphasis (SRLGE)
- short run high grey-level emphasis (SRHGE)
- long run high grey level emphasis (LRHGE)
- long run grey level emphasis(LRLGE)

The first two were introduced by Chu et al [53] and the other 4 features were introduced by Dasarathy et al [54]. These features not only use the number of runs but also the grey level values associated with them (make use of the distribution of the grey level runs). Chu et al [53] admits that these features are not to replace the corresponding classical GLRLM features as cases can be constructed such that the latter outperform the new features. They exploit the idea that the distribution of runs and the distribution of grey levels are distinct entities. This is useful in dealing with images which have the same run length but different grey level distribution. This is what makes these features distinct from the corresponding features in the classical GLRLM.

The SRHGE and LRLGE characterise the coarseness of the surface of texture. High SRHGE and high LRLGE mean fine and coarse texture respectively. The low grey-level run emphasis (LGRE) and the high grey-level run length emphasis (HGRE) characterise are measures of coarseness of the textured surface. These features were not pursued further as in this context they did not shown any improvement and furthermore they add to the computational complexity of the system.

Feature	Measures	Good Paper Surface
SRE	fineness	high
LRE	coarseness	low
GLNU	distributions of runs	low
RLNU	distributions of runs	low
RP	percentage of runs	low

Table 3.1: *This is a summary of the GLRLM matrix features*

In this thesis, the directions of the selected runs were 0° and 90° because the surface appearance features tend to be distributed in the machine and cross directions. The weaknesses of the GLRLM are that it does not consider grey level transitions which are key to assessing uniformity of a surface. Additionally it is sensitive to noise. Tang obtained 88% classification

performance using new GLRLM features introduced by CHU et al [53] and Dasarathy et al [54] in addition to the traditional GLRLM features [55] on "Vistex" images that include fabric, metal and grass among others and also on "Brodatz" images that include rice, paper and straw.

The application of GLRLM has been in analysing beef muscle tissue [56], in beef grading [57] and in natural textures [58] among others.

3.3 The spatial grey level dependence matrix (SGLDM)

This section introduces the spatial grey level dependence matrix sometimes called the *grey level co-occurrence matrix*. What makes this technique different from others is its transition probability matrices.

3.3.1 Introduction

The spatial grey level dependence method (SGLDM) is derived from a count of the occurrence of two grey level values at a separation d and in the direction θ [45]. This is the probability $P(i, j, d, \theta)$ of going from grey level i to grey level j given the inter-sample distance d and the direction θ. This transition probability makes the SGLDM unique from other techniques.

Figure 3.2 is an illustration of feature extraction from a test image using the co-occurrence matrices.

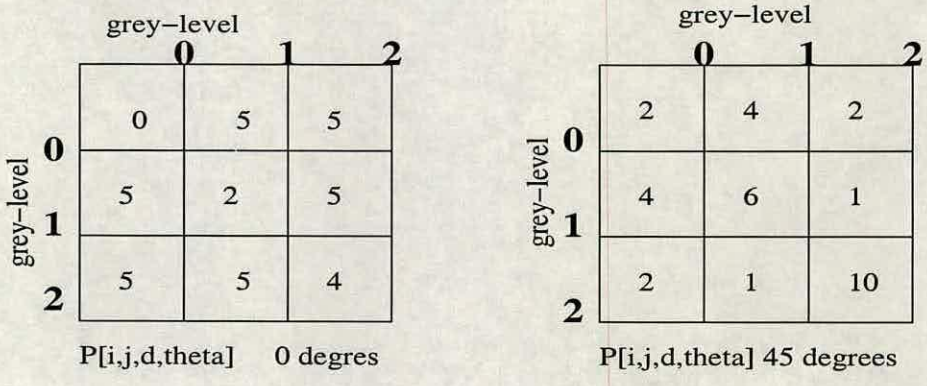


Figure 3.2: *The co-occurrence matrix results shown in this table were computed from the test image shown in Figure 3.1*

Thus the SGLDM is dependent on spatial relationships in the grey levels [45] and on the re-

gional intensity background variation.

By varying parameters θ and d this method can be calibrated to a range of different textures. The procedure used to attain the optimal values for θ and d are presented in chapter 7 in the section on optimisation. The procedure is repeated for all possible pairs of grey level values in the image. The size of the resultant SGLDM matrix is determined by the number of distinct grey levels.

Having calculated the matrix for an image its "content" may be characterised using statistics. Those that are typically extracted from this method are: entropy, homogeneity, energy, contrast and absolute value. These are calculated as follow [8, 45]:

$$\text{Contrast} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}(i-j)^2 P_{(i,j,d,\theta)} \tag{3.13}$$

$$\text{Entropy} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} P_{(i,j,d,\theta)} \log P_{(i,j,d,\theta)} \tag{3.14}$$

$$\text{Absolute value} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} |i-j| P_{(i,j,d,\theta)} \tag{3.15}$$

$$\text{Energy} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} P^2_{(i,j,d,\theta)} \tag{3.16}$$

$$\text{Correlation} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}(i-\mu_i)(j-\mu_j)\frac{P_{(i,j,d,\theta)}}{(\sigma_i\sigma_j)} \tag{3.17}$$

where μ_j and σ_j (10) are the mean and standard deviation of the column of the SGLDM matrix respectively, whilst μ_i and σ_i is the mean and standard deviation for the rows, respectively.

The correlation feature is a measure of the grey level linear dependencies in the image. A fine structure has a small correlation value because adjacent pixels are less correlated.

$$\text{Homogeneity} = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\frac{P_{(i,j,d,\theta)}}{(1+(i-j)^2)} \tag{3.18}$$

$P(i,j,\theta,d)$ is the matrix of relative frequencies, relative because of $i-j$, frequencies because of the repeated occurrence of $i-j$. The matrices are *symmetric* meaning that the $(i,j)^th = (j,i)^th$ element.

These features can be used to assess image texture, for example, contrast is a measure of the degree of local variation in an image. If i and j are similar, then from the operation (i-j), the contrast value will be very small which indicates uniform paper. The SGLDM is a matrix of relative frequencies (i-j). In the case of entropy, a large probability results in zero or near zero entropy since the log of 1 is 0. Entropy measures the distribution of spectral components in an image. High entropy indicates non-uniform texture, for example, texture might contain many edges. The level of entropy that constitutes good paper the co-occurrence matrix whose probability is within the range of 0 to 0.4. The probability above 0.4 is viewed as that representing a non-uniform surface.

Good quality paper is also characterised by a high energy statistic. Energy is thus directly proportional to the probability. A good paper surface has all the probability mass concentrated in one histogram bin. When the probability is high, energy will be high. The (i-j) term in the homogeneity equation will result in a small denominator and thus a high homogeneity value. The SGLDM is dependent on spatial relationships in the grey levels [45] and on the regional intensity background variation [45]. In terms of weaknesses, the SGLDM is ineffective in characterising relatively random, low contrast textures [59]. This is because it is based on transition probabilities since low contrast means very low variability (or transitions) within a chosen neighbourhood on the surface of paper, hence different paper characterised by low contrast could result in almost equal feature values hence discrimination becomes hard.

The SGLDM has 14 features described by Haralick [45] but the five commonly used SGLDM measures that include energy, entropy, correlation, homogeneity and contrast might not always contain all the important texture-context information contained in the SGLDM [8]. The texture-

context information is adequately specified by this matrix and visually distinct texture pairs can be discriminated using the SGLDM [60]. This is because apart from containing information of transition between pixels at a given distance it also includes the direction information. Since we cannot use this matrix as an input to the classifier and that this matrix has 14 known features, it is therefore profitable to compute only a few intuitively selected features that are supposed to capture a large percentage of the information that this matrix contains. The SGLDM does not consider primitive shapes and therefore should perform poorly on texture composed of large primitives [25]. However, texels from paper images are small thus this should not pose a problem.

In terms of application, Julesz [9] was the first to use co-occurrence statistics in visual human texture discrimination experiments. Recent applications in related work have been in land cover classification [61], in terrain Classification [11], in visual inspection [62], in the characterisation of liver tissue [63] and ultrasonic liver images [8, 64] and in detecting abnormalities in medical images [47, 48] and in analysing beef muscle tissue [56] and in beef grading [57]. These applications are in addition to the ones described in detail in section 1.2.1.

Southard et al [51] found SGLDM features useful in detecting osteoporosis (a condition in which bones become brittle). Their objective was looking for texture features which change most with minimal bone loss. It turned out that the SGLDM angular second moment and entropy were correlated with calcium loss.

The SGLDM's dependency on the number of grey levels in the entire image [64, 65] is a limitation. As an example, let an image have 2000 zeros in the image with the rest of the pixels taking values of 3, 7 and 63. On constructing the co-occurence matrix, more time and computing resources will be wasted in computing zeros. In addition the SGLDM needs to be recomputed should an image's dynamic range change. However, memory requirements are no longer as limiting as computers with large memory storage are now available. The SGLDM works well on a variety of textures [66] and it has been found to give good overall performance [45] on textured images. The optimal angles and distances chosen for a co-occurrence matrices which are given in chapter 7 should result in an SGLDM that is sensitive to the underlying structure contained in the texture.

The second order statistics approach to texture analysis is useful as it considers the spatial relationship between neighbouring pixels and this is intuitively appealing as it is in agreement

with the definition of texture given in chapter 2 (texture is a neighbourhood property).

3.4 SGLDM Summary

The decrease in energy corresponds to a spreading of the grey level transitions in the SGLDM. The opposite directions of the co-occurrence matrices were ignored because of the symmetry. Weszka et al [11] [67] found that small values of inter-sample distance d yield the best results on terrain classification.

The attraction of the SGLDM is its ability to describe texture and its power as a tool for discriminating different textures. However, it is inefficient in terms of memory requirements. The SGLDM is based on repeated occurrence of some grey level configuration (grey level transition) in the texture which might vary:

- rapidly with the distance in fine textures,
- slowly with the distance in coarse textures [45].

The windows (size of neighbourhoods) for computing the SGLDM need to be optimised. The optimisation of these windows is covered in chapter 7. Larger windows are computationally expensive, they tend to give smoother features, consequently a lower classification performance [59] is obtained.

Texture is coarse when the inter-sample distance d is small because the pixel pairs at separation d have almost similar grey level values and the SGLDM matrix values are concentrated on or near the diagonal. Conversely, for a fine texture, if d is comparable to the texel size, then the pixels separated by d are spread out uniformly.

A uniform texture has few entries corresponding to small transitions or no transitions around the leading diagonal of the SGLDM matrix. Thus paper is uniform when the texels are small and the texel frequency is high. A nonuniform texture has entries further apart from the diagonal and a high contrast value and this spread in grey level transitions results in the loss of uniformity. Uniformity is therefore a function of the distribution of the texels on the surface of paper.

In terms of paper, homogeneity is a measure of evenness of the texture on the surface of paper. It emphasises the contribution of diagonal entries. High and low homogeneity values represent

good and poor quality paper surface respectively. Homogeneous paper surface has a surface with texels (texture elements) of the same size. If the sizes of the texels increase with no significant increase in density, then the surface becomes non-homogeneous. The higher the texel density, the more homogeneous the surface is. Thus homogeneity is a function of the texel densities. Thus density is what distinguishes homogeneity from uniformity.

Entropy measures the randomness (the degree of disorder) of the distribution of texels on the surface of paper. Thus entropy is a measure of non-uniformity. Low entropy is when there is only one entry in the SGLDM matrix and it represents a uniform texture and in this context good quality paper surface. High entropy means a large number of grey level transitions and hence poor quality surface appearance.

From the definition of uniformity above, it can be seen that both entropy and homogeneity are measures of uniformity.

A texture is directional if it shows a progressive increase or decrease of coarseness as we move away from a point on the image. In this case the direction θ can be 0°, 45°, 90° and 135° among others. Thus the degree of spread about the SGLDM matrix's main diagonal varies with θ hence computing spread for various directions of d is useful. For a typical image profile, nonzero matrix elements are dispersed along or near its main diagonal.

Feature	Measures	Good Paper Surface
Entropy	non-uniformity	low
Contrast	non-uniformity	low
Absolute value	uniformity	low
Correlation	grey level linear dependence	high
Homogeneity	homogeneity	high
Energy	homogeneity	high

Table 3.2: *This is a summary of Co-occurrence matrix features*

3.5 The Grey Level Difference Method

The grey level difference method (GLDM) [8, 11] is a matrix of the frequencies of pixel pairs which have a predetermined absolute difference in grey level and which are separated by a displacement d.

Figure 3.3 is an illustration of feature extraction using the GLDM feature extraction technique.

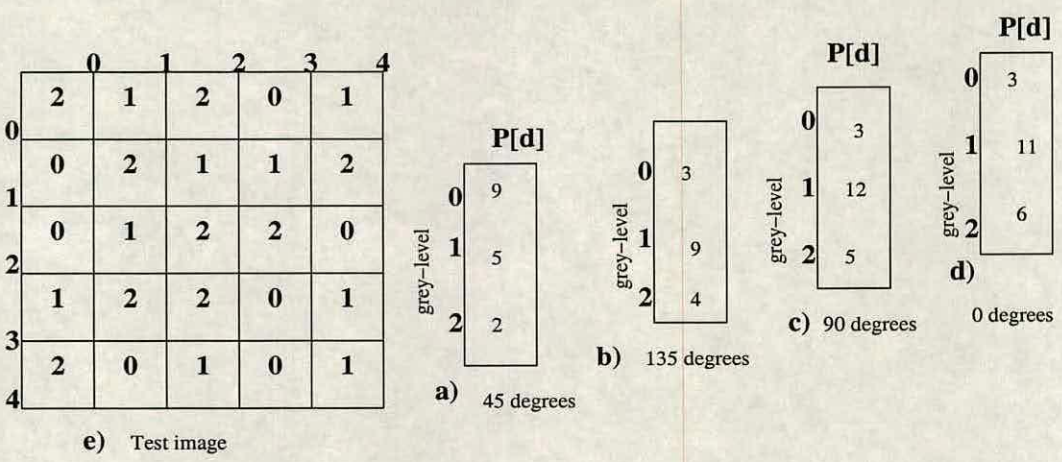


Figure 3.3: *The distance d for this GLDM matrix is* 1. *(a) angle* 45°. *(b) angle* 135°. *(c) angle* 90° . *(d) angle* 0°. *(e) Test Image*

The GLDM is given by:

$$I_{\sigma} = |I(x, y) - I(x + \delta x, y + \delta y| \tag{3.19}$$

The displacement $\sigma = \delta x, \delta y$. Its implementation involves counting the number of times each I_{σ} occurs. A probability density function P_{σ} is formed and if there are m grey levels, P_{σ} is an m dimensional vector, and in this work it is a 256 dimensional vector.

The GLDM is based on the assumption that useful texture information can be extracted using first order statistics from the GLDM matrix.

The features computed from the GLDM are the inverse difference moment (IDM), angular second moment (ASM), mean, entropy and contrast. The angular second moment (ASM) feature is a measure of homogeneity of the image. It is the sum of square of the entries in the matrix. In a homogeneous image there are very few dominant grey-level transitions. Thus the matrix will have fewer entries of large magnitude.

The IDM feature is at times called homogeneity. It measures image homogeneity as it assumes larger values for smaller grey-level differences in pair of pixels. It is inversely related to contrast

and energy. IDM is a texture parameter affected by both textural frequency and contrast The moment of inertia about the origin is also called the second moment of P_σ. The GLDM can be used to characterise coarse texture if the displacement σ is small compared to the texel size, consequently a small $I_\sigma(x, y)$ results and P_σ will be concentrated near zero. The mean feature is small when P_σ(i) are concentrated near the origin. Texture is fine if P_σ is more spread out resulting in a large $I_\sigma(x, y)$ value and the pixels separated by d which is comparable to the texel size will be quite different. Entropy is largest for equal P_σ(i) and small when they are very unequal. In contrast, coarse texture images have low values when d is small compared to the texel size. This is due to pixels having a similar grey level value, consequently, P_σ will be concentrated near $i = 0$.

The σ can be varied and spread of values in P_σ noted. A small d results in most data concentrated near zero while for a larger d they will be more spread out. Thus the comparison of spread measures for various directions of d is key to the measure of texture directionality. In terms of paper surface appearance quality assessment, entropy must be a useful measure [11]. Weszka et al reports that contrast feature performs better than the IDM features [11]. In the context of characterising the surface appearance of paper, this is likely to hold because contrast unlike IDM is based on the difference in pixels values which is a measure of variability.

The angular second moment is smallest when the data values are equal otherwise they are large and histogram values tend to be higher for nearly equal pairs [11].

What follows is the relationship between the GLDM and the SGDM.

$$P_\sigma = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} p(i, j, d) \tag{3.20}$$

where $|i - j| = k$, $k = 0, 1, 2, ..., N$, and p(i,j,d) is the (i,j)th element of the co-occurrence matrix. P_σ the probability density. k is the grey level difference between pixels separated by d.

Thus P_σ can be obtained from the SGLDM by summing up the entries along axes parallel and symmetrical with respect to the main diagonal of the matrix. Consequently, features derived from the SGLDM are similar to features derived from the GLDM matrix, with contrast feature being the same for both [11].

The related application of the GLDM has been in characterising liver images [64] and ground

Feature	Measures	Good Paper Surface
ASM	homogeneity	low
CONTRAST	non-uniformity	low
ENTROPY	distribution	high

Table 3.3: *This is a summary of the GLDM matrix features.*

cover identification in satellite images [68].

The GLDM is less computationally expensive than the SGLDM [64]. For example, the GLDM in a 256×256 image forms a 256 dimensional vector whereas the SGLDM forms a 256×256 matrix to describe the same texture surface. However, since the goal is not real time implementation but a compromise between accuracy and speed, then the SGLDM is preferred to the GLDM. Whereas these techniques tend to give almost similar results on some problems, the most reliable and the the one that is commonly used by other workers in the field is the SGLDM.

3.6 The Neighbourhood grey level dependence matrix

The neighbourhood grey level dependence matrix (NGLDM) [46] is a matrix that solved the angular dependence and computationally complexity of the SGLDM. The implementation of the NGLDM involves counting the number of times a neighbouring pixel occurs, at a distance **d** and a predefined grey level difference **a** from the centre pixel. Figure 3.4 is an example of feature extraction from a test image using the NGLDM matrices.

The NGLDM hereafter called the **Q** matrix captures information from texture. **k** is the row of the Q matrix whilst **s** is the column of the Q matrix. Thus **Q** is the number of times the difference between each element $f(i, j)$ and its neighbours is equal to or less than a certain distance **d**.

Each entry in the NGLDM matrix represents the number of occurrences of the pair (**k,s**). An example is $(d = 1, a = 0)$. A $d = 1$ implies 1 pixel on either side of the current pixel which translates to a neighbourhood of 3×3. For a 3×3 neighbourhood, **s** takes values of 0 to 8 inclusive.

What distinguishes it from the SGLDM is that the current pixel and all its neighbours are

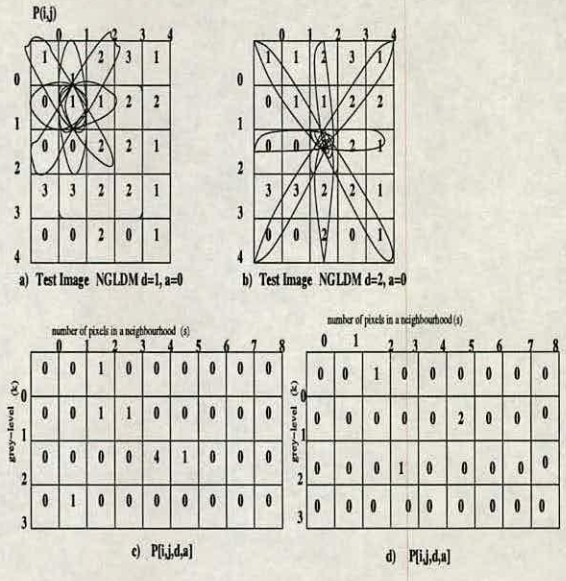


Figure 3.4: *Figure 3.4 a and b are Test images and the loops on them are an illustration for the distances d =1 and d = 2 respectively. Figure 3.4 (c) and (d) are the NGLDM matrices for distances d = 1 and d = 2 from Figure 3.4 (a) and (b) respectively.*

computed at one time instead of computing in one direction at a time. The current pixel is excluded from the computation.

The degree of texture coarseness of an image is depicted by the distribution of the entries in the **Q** matrix. However, because of its size, it is expensive to use it directly as an input to a classifier, instead, features that describe texture are extracted from this matrix. These features include the*small number emphasis* (SNE), the *number nonuniformity* (NNU), *large number emphasis* (LNE), the *second moment* (SM) and *entropy* (ENT). These features are invariant under spatial rotation and linear grey level transformation. Their insensitivity to monotonic grey level transformation is increased by manipulating **a**.

Changing **a** and **d** can alter the distribution of the NGLDM numbers because coarseness and fineness of an image are not absolute hence optimal values for these parameters have to be found.

The SNE is a measure of fineness of the image and a fine textured image is characterised by entries with large values in NGLDM's low index columns making $Q(k,s)/s^2$ term in Figure 3.4 large for small **s**, and this occurs in many situations where neighbourhood similarity is small.

LNE is a measure of coarseness of an image and a coarse textured image is characterised by large NGLDM numbers concentrated in the large s columns making the $s^2Q(k,s)$ term in Figure 3.4 larger for large **s** and this implies in a lot of situations where neighbourhood similarity is large which is characteristic of coarse texture or total homogeneity.

The Second Moment (SM) measures the homogeneity of the **Q** matrix. In a homogeneous image the **Q** matrix has only a few large number entries . As a consequence, SM will be large because it is the sum of the squares of all entries in the **Q** matrix. The NNU and entropy are related to coarseness of an image [13].

Feature	Measures	Good Paper Surface
SNE	fineness	high
LNE	coarseness	low
SM	homogeneity	high
NNU	non-uniformity	high
ENT	non-uniformity	low

Table 3.4: *This is a summary of the NGLDM matrix features.*

3.7 Chapter Summary

The FOS is based on a count of the number of pixels with the same grey level value (the occurrence probability of intensity) in being placed in one bin. The FOS does not consider the spatial inter-relationships between the pixels and hence the structure of the surface information is lost. However, its variance feature has been useful in characterising paper although there was no constraint put on the feature sizes of interest.

The GLRLM [52] assesses the uniformity of the grey level values of adjacent pixels in an image. It computes the number of times an image contains a run of a given length for the current grey level. Features are then extracted from this matrix. In a course or fine texture long runs and short runs dominate respectively. The length of the runs reflects the size of the texels. However, the GLRLM does not consider grey level transitions which are key to assessing uniformity of a surface. Additionally it is sensitive to noise.

The SGLDM [45] incorporates direction of texture and transition between pixel information at a given distance and thus it is suited to this task. The SGLDM is dependent on spatial relationships in the grey levels [45] and on the regional intensity background variation. However, the

SGLDM performs poorly on relatively random, low contrast textures [59] because of very low grey level transitions within a neighbourhood.

The GLDM [8, 11] matrix's entries are a count of the number of times each grey level difference occurs. Useful GLDM texture information is extracted using first order statistics. The GLDM can be obtained from the SGLDM, consequently, features derived from the GLDM and SGLDM matrix are similar [11] yet the former is less computationally expensive than the latter [64]. However, since real-time implementation is not feasible not useful as the paper has to lose moisture before it can be assessed for quality, a compromise between accuracy and speed is key. Thus, SGLDM is preferred to the GLDM as the former is most reliable and most commonly used by other workers in the field.

NGLDM was introduced as a technique that was to solve the angular dependence handicap of the SGLDM. The degree of texture coarseness of an image is depicted by the distribution of NGLDM matrix entries. The current pixel and its neighbours are computed at one time (direction independent) and not one direction at a time as happens with the SGLDM. However, in papermaking the direction information is useful as it potentially assists in identifying problems that may have arisen from the machine that produces paper.

.

Chapter 4
Spectral Techniques

4.1 Spectral Techniques

This chapter presents an overview of spectral techniques which include the discrete Fourier Transform, the discrete wavelet transform, the discrete wave frame transform, the Gabor transform and the discrete cosine transform. These transform techniques are widely used in image processing and image analysis and they must be useful in paper surface inspection as well.

Transform coefficients are weighted sums and differences of pixel values.

4.1.1 Chapter Introduction

The aim of this chapter is to assess different spectral techniques' performances when applied for textured surfaces. Their potential application on paper will be explored.

The analysis of images in the spatial domain is commonly used although it does not always give a full description of the image. The distributed spectral components are difficult to detect using spatial methods. There is therefore a need to represent information contained in the texture in a domain where it is easily described. This is achieved through the use of transform techniques that analyse the frequency content of an image. These techniques, called spectral methods break up the image into its constituent spatial frequency components. The discriminatory information is extracted from these frequency components (coefficients). A common spectral feature, "energy", is unevenly distributed in the frequency domain, for example, in the Fourier spectrum, it generally decreases from the zero frequency to the high frequency end of the spectrum. Consequently, sub-spectra with more energy (for example in wavelets) content are retained for use in later stages like classification whilst the rest are discarded.

4.1.2 Fourier Transform

The Fourier transform (FT) breaks up a textured image into its constituent spatial frequency components. The images used here are digital images and thus the focus is on the discrete Fourier transform (DFT). Any feature of the DFT that is not important to texture classification has been deliberately left out. The following is the relation for the DFT:

$$\text{F(u,v)} = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} f(x,y) e^{[-j2\pi(ux+vy)/N]} \tag{4.1}$$

In (4.1) F(u,v) and f(x,y) are the Fourier Transformed image and the image respectively. The implementation of the DFT involves convolving the image with the DFT kernel. The 2D-DFT convolution, because of the 2-D DFT separable kernel, it can be accomplished by computing a 1-D DFT of each row resulting in an image f' and then followed by a 1-D DFT of each column of f'. The benefit from separability is an increase in the 2-D DFT computation speed.

The discrete Fourier transformed image has the following features:

- The frequencies of the basis functions increase from the lowest (DC) to the highest frequency (AC) of the Fourier image.
- The DC component is at the top left corner of the transformed image.
- The DC component gives average brightness.
- The AC component increases from DC and its highest value is at the bottom right corner of the transformed image.
- The background noise gives the average background.

The advantage of the Fourier domain image over the spatial domain image is the former's large dynamic range (values stored in float). The amplitude and not the phase (the phase is important for reconstruction which is not the issue here) is useful for this work hence less computation time results. This is from (4.2) where the Fourier transform of a signal that has been translated by "a' ' results in a shift in the phase and not in the power spectrum (amplitudes) [30]. This shift invariance of the amplitude is integral in image recognition.

Noise in the DFT although it manifests itself as a broad peak it does not affect the periodicity of the signal. The only loss in the DFT is in the precision from rounding and truncation errors.

$$FT(f(x-a)) = F(w)exp2\pi jwa \tag{4.2}$$

The Fourier amplitude and power spectrum (energy peaks) are given by $|F(u,v)|$ in (4.3) and $(|F(u,v)|^2)$ in (4.4) respectively.

$$\text{F(u,v)} = R(u,v) + jI(u,v) \tag{4.3}$$

$$\text{F(u,v)} = [\mathbf{R^2(u,v) + I^2(u,v)}]^{\mathbf{1/2}} \tag{4.4}$$

R(u,v) and I(u,v) are the real and the imaginary components respectively. These peaks are global texture patterns and their distribution gives the directionality and global periodicity information of the texture patterns. Peaks near or further from the origin represent a surface consisting of coarse and fine texture respectively. In a periodic texture anything that comes between the peaks is a defect. However, peaks require skill for their interpretation and their inspection consumes a lot of time.

Another important feature from the Fourier transform is energy extracted from the spectrum. Energy is defined as the total sum of the square of coefficients of the Fourier transformed image. This feature would be a very effective if it took advantage of the periodicity and the distribution of spectral components.

Most of the energy in a Fourier transformed image is in the DC components, however, unlike in the case of data compaction where high frequency components are discarded (since most of the energy is in low frequencies), for texture classification high frequency coefficients are retained as they also contribute to the accuracy in classification. Otherwise using energy compaction as a criteria for comparison, the spectral techniques outclass the spatial methods as in the latter, patterns are difficult to detect due to its neighbourhood approach.

The DFT has also some disadvantages. A local characteristic, like a discontinuity in the signal illustrated by (4.5) becomes a global characteristic of the transform and thus it cannot be located

precisely. Furthermore, it corrupts the entire transformed image as information is dispersed throughout the frequencies of the entire transform. In (4.5) from Gonzalez [30], x is the size of the discontinuity and v is the frequency.

$$\delta x = \frac{1}{\delta v} \tag{4.5}$$

This is also typical of the Gibbs' phenomenon, a consequence of the truncation of the DFT coefficients and shows up as dominant peaks at these discontinuities. The focus of this work is not on a defect or its location, but on the change in the distribution of "ridge/valley" appearance on the paper. Short duration signals often carry interesting information, however, the Fourier transform performs badly on these. This is another handicap.

Computing the FFT using DFT might result in aliasing [30, 69], a phenomenon that occurs when high frequencies in a continuous signal become low frequencies in the digitised signal. It is a consequence of a signal that has been sampled at less than twice its highest frequency (Nyquist) in each cycle and as a result contributions from adjacent periods tend to corrupt the signal. Another phenomenon, the picket-fence effect, occurs due to frequencies that are not an integer multiple of the fundamental frequency. There must be an integer multiple number of cycles in an image otherwise when it is Fourier transformed, leakage will result. A sampling interval δx should be chosen such that $1/2\delta x \geq W$ to avoid overlap. W is the bandwidth. The δx can be decreased to separate the periods [30]. Aperture effects (spurious frequencies at the edges of an image or at a defect) are a weakness in performance by Fourier features compared to gray level statistics [45] and those based on FOS of GLDM Weszka et al [11].

A spatial sampling rate of 8pixels/mm and a neighbourhood of 10×10 mm was used by Lois M. Hoffer et al when he used Fourier transform for quality control of cloth on a loom [70]. He divided this cloth into 5 classes according to different sizes of defects. Claudio et al successfully used Fourier transform on fabric [71]. First order spectral features in conjunction with textural features have been used by Haralick [7]. The other application of the DFT are in image filtering, image reconstruction and image compression.

In a comparative study, Conners concludes that the power spectrum method has the least performance when compared to SGLDM, the GLDM and the GLRLM [8]. The DFT is efficient in

computation and has good energy compaction (concentrates information in a few coefficients) properties. The DFT performs better than other spectral techniques (not spatial techniques) in most pattern classification applications.

4.1.3 Summary

The assumption by the Fourier Transform that signals are periodic is a major weakness as this assumption does not always hold. In addition, the Fourier Transform in decomposing the image uses a window of infinite width which is integrated over all space hence any point in space where the frequency appears affects the result equally. The Fourier Transform thus describes the global frequency content of an image yet texture is a local property. "Global" because its basis functions (sinusoidal signals) do not have compact support and thus a loss of spatial information occurs as in chapter 2 it was established that texture is a neighbourhood property. As a consequence, the Fourier Transform is incapable of capturing the local variation in texture which is what is typical of most textured surfaces.

There is periodicity in the pattern in the cross direction (the direction perpendicular to the motion of the conveyer belt) of the paper due to the wire mesh used in the early stages of paper production. Thus the Fourier transform is recommended only in dealing with the cross direction. Since in this work we are dealing with the machine direction, the surface appearance of paper is random hence the Fourier Transform method might not be suitable. Additionally, the DFT is the sampled version of the Fourier transform and therefore does not contain all frequencies contained in an image, but only enough samples to capture information from the spatial domain image. Furthermore, the DFT does not provide sufficient accuracy when applied to small samples. The DFT has been included here because even random textures have some degree of periodicity.

4.1.4 The Windowed Fourier Transform (WFT)

The WFT [72] commonly known as the short-time Fourier transform allows frequency and space information to be displayed together [73]. The shortness is in the finiteness of the window (compact support). This transform in addition to identifying the frequency components present in the signal, it also finds their location in space.

The implementation of the WFT involves splitting a non-stationary signal into segments (win-

dow widths) which are assumed to be stationary within the interval of one frequency component. A Fourier Transform of the product of the window function and the image gives the frequency of the data at a point. The subsequent frequency components are extracted by shifting this window and repeating the process. Any changes in the frequency content of an image shows up in the resulting 2-D WFT frequency plot.

The long WFT duration window eliminates impulsive noise in the spatial domain and also localises the sinusoidal signal because of averaging and other frequencies just get averaged. This technique does resolve the spikes in space, but fails to detect rapid frequency changes. Striking a balance between the two is called the Heisenberg inequality [72] which states that a signal does not simultaneously have a precise location in space and a precise frequency.

In summary, the WFT, unlike the FT, its frequency resolution is suboptimal because its window is of finite length. The wavelet transform easily performs this task but the basis has to be set correctly.

4.1.5 Multiscale Texture Analysis

4.1.5.1 Introduction

This section presents multiresolution techniques. They are spectral techniques that decompose an image at different scales (resolution level) and the information contained in them is represented as a collection of subimages of a specific scale and orientation. The assumption is that information contained in texture resides in some scale in an image. Since the exact scale may not be known, a range of possible scales must be explored.

The motivation for these techniques is their ability to mimic the human visual system [74] which processes images in a multiscale way. The notion of scale might be useful in texture classification but for paper quality it might be insignificant as the scale of the texture is more tightly defined. The multiresolution techniques that are discussed in this section include the discrete wavelet transform (DWT), the discrete wavelet frames (DWF) and the Gabor transforms.

4.1.5.2 Wavelets

The term wavelet [75] refers to a small wave and this smallness is in the amplitude of the wave which decays to zero quickly with distance. The basis set for wavelets generated by

the mother wavelet kernels have different support (different resolution levels) and thus during decomposition they adapt to spectral components in the image. In wavelets the key is the notion of successive approximation which views a signal as a "coarse" version plus added "details". This is intuitive and the details on how it works is covered in this section.

The discrete form of a wavelet is commonly used because the image is in digital form. The wavelet has a scale-translation domain. It has orthogonality of translates (the window is translated across the image) which results in a highly decorrelated output. Discrete wavelet decomposition involves shifting the window in space (localising the signal in space and spatial information in the transform domain) and then convolving the image by a DWT kernel for each shift. The results from this decomposition are four subimages, the vertical, horizontal, diagonal and the approximate image. Each of these subimages contains spatial information of a specific scale and orientation. This linear expansion [76] of the image or scaling is viewed as a correlation between the wavelet at different scales with the DWT kernel. Non-zero coefficients occur at the transition points (a change from one frequency to another) and they show up as ripples in the spectrum.

The next step involves subsampling otherwise an overcomplete representation results. The process is repeated on the first level image "L" Figure 4.2 or Figure 4.1(a) to get third level Figure 4.1(b) and iteratively on subsequent low pass subimages until a certain criterion is met. Further decompositions either enhance or degrade detail. The original image is called the base level image. The pyramid method increases the variance of the image. Higher images (towards the peak of the pyramid) have more variance than those at the base. At the base, the analysis of the image is global analysis of texture whereas higher up the pyramid, the analysis becomes more local as subimages become smaller and thus more fine detail can be captured from the image. This fine detail is the "more variance" . As an illustration, the base level image is an analogue of a person viewing an object at a distance whilst images higher up the pyramid is the same person now viewing the object at close range. Thus in the former case, less detail is seen (blurred vision) than in the latter case (crisp image). Variance is a key feature in texture classification.

The DWT has a good spatial resolution and a good frequency resolution for a narrow kernel and a wide kernel respectively. This is characteristic of signals encountered in practice. The DWT has a poor spatial and frequency resolution for a wide kernel and a narrow kernel respectively.

A wavelet is viewed as a high and low pass filter which gives a low and high pass detail image respectively [77]. What makes the DWT unique from other spectral techniques is the aspect ratio of the window (support) which changes whilst the area under the window remains constant. Additionally, the DWT is computationally inexpensive and less complex than other wavelet forms. The relation of a wavelet is:

$$f(x) = \sum_{\nu_{finite}} \sum_{k} finite C_{\nu} \Psi_{\nu k}(x) \tag{4.6}$$

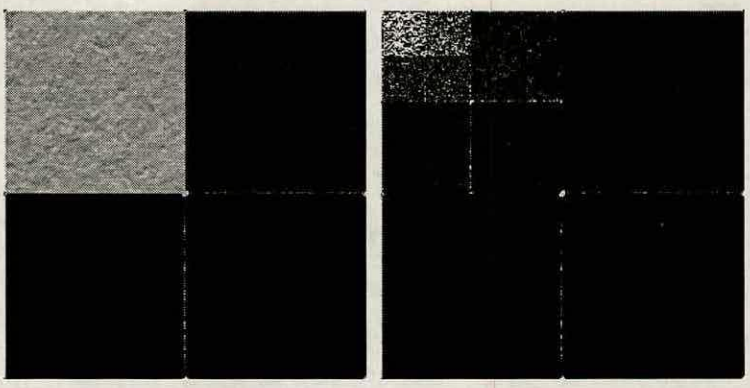
where $\Psi_{\nu k}(x) = 2^{\frac{\nu}{2}} \Psi(2^{\nu} x - k)$. This alters the window size and the function is translated over integer values k which in turn shifts the energy localisation to the next point on the image. This integer translation defines the family of basis functions. The scaling of the variable x helps to increase the limited space spanned by the wavelet equation. The $2^{\frac{\nu}{2}}$ is for normalising the basis. All the basis functions in Ψ_i are scaled and translated versions of the mother wavelet (Ψ_i) and the translation over the image in steps of size $2^{\nu}k$. A linear combination of all these step sizes gives a wavelet decomposition of the signal. The scaling factor ν in $\Psi(2^{\nu}x)$ is a power of 2 which gives the desired cascaded octave bandwidth filter structure since the bandwidths of the frequency of the decomposed signal and centre frequencies must vary by octaves. The $C_{\nu k}$ are the coefficients computed by the wavelet transform.

The wavelets have some disadvantages as the resolution in space and frequency cannot be made small because the space-bandwidth product given by:

$$\Delta x \Delta f \geq \frac{1}{4\pi} \tag{4.7}$$

where Δx represents space and Δf represents the bandwidth. The wavelet is also limited by the Nyquist sampling rate [30].

The DWT decomposition of paper images is shown in Figure 4.1. These images show a lot of energy in the DC component of every level (the top left corner image for each level). It is evident that the largest component of the original image is in the DC. The rest of the subimages are darker which means that they contain a small fraction of the image information.



(a) The "good" paper image has been transformed using level 1 of a DWT. This image shows a bright image at the top left corner and a very dark image at the bottom right corner

(b) The "good" paper image has been transformed using level 3 of a DWT. The bright image at the left top corner is the (DC) and a gradual decrease in brightness towards the bottom right corner is the (AC) part

Figure 4.1: *The* Discrete wavelet transformed *of paper images*

The sum of wavelet coefficients is zero and thus a square of the coefficients is taken before summing them up and the result is a feature called energy. If the energy is in high frequencies, the localisation of these frequencies in space is good since the signal is characterised by many samples. The low scales are the detailed part of the signal and are better resolved in space. In contrast, high scales (global non-detailed) are better resolved in frequency and they give large values for the entire duration of the signal since low frequencies (height is short, width is wide) always exist. The energy feature computed from wavelet coefficients is given by:

$$\text{energy} = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} |x(m,n)| \tag{4.8}$$

where x(m,n) is the decomposed image. where $1 \leq m \leq M$. Wavelets overcome the resolution limitations in frequency of the Windowed Fourier Transform (WFT) due to the former's variable-width windows and thus information in different resolution levels is captured. A wide window (the space-frequency plane) leads to good frequency resolution and poor spatial resolution. A narrow window enhances the detailed properties of the image and leads to a good spatial resolution. Wavelets therefore possess almost all the characteristics of the Fourier Transform in

addition to the former's localisation property. The wavelets are therefore preferred to 2-D FT in texture analysis. Image detail is quasi-stationary and the amount of detail in any subregion depends on the spatial location.

Wavelet application areas include signal and image de-noising, in medical diagnosis [78], image enhancement and image compression. One version of wavelets which performs better than the DWT is the discrete wavelet frame (DWF).

4.1.5.3 The Discrete Wavelet Frames

The Discrete Wavelet Frames (DWF) [72] is a redundant overcomplete wavelet. The overcomplete property means that it is not subsampled, consequently, it retains all the energy.

This is the relation for the frame:

$A||f(x)||^2 \leq \sum_{k\in J} | < \phi_k, f(x) > |^2 \leq B||f(x)||^2$, f(x) is a signal, k is an integer index of a finite sum. A and B are frame bounds. A family of functions ϕ_x in Hilbert space H is called a frame if there exists $A \geq 0$ and $B \leq \infty$ so that for all $f \in H$. $A \geq 0$ means that the output will not be null if the input is non-zero, $B \leq \infty$ means that given an input of finite power, the output power to transform has to be finite also. $\frac{(A+B)}{2}$ is a measure of redundancy of the frame.

A tight frame is a special frame with equal bounds (A = B) and it provides a uniform gain for the frequency response of the input function. $\frac{B}{A}$ is a measure of the tightness of the frame. The tightness of frames depends upon how the energy of the signal is preserved by the decomposition.

DWF are superior to the critically sampled (subsampled) wavelet transform feature extraction [72].

In terms of texture description, the DWF is invariant with respect to translations of the input signal. It decreases the variability of texture features resulting in better classification performance [72]. A multiscale feature extraction of more than one level gives higher discrimination than a single resolution analysis [72]. The performance of a single scale DWF has been found to match that of local linear transform in feature extraction.

The relation for the DWF:

$$f(x) = \sum_i c_i \Psi(x) \tag{4.9}$$

Where $\Psi(x)$ are the basis functions and c_i are the coefficients. Since the former are fixed, it is the latter which contain the information about the signal. We want to know the frequency content of a signal. Wavelets give an instant in space in which this frequency occurs. However, the resolution in space δx and the resolution in frequency δw cannot both be made arbitrarily small at the same time because their product is lower bounded by the Heisenberg inequality [72].

$$\delta x \delta w \geq \frac{1}{2} \tag{4.10}$$

Thus it is only possible to get a good resolution in space in return for a low resolution in frequency, and vice-versa. A compromise for the Heisenberg inequality can be achieved through the use of a set of basis functions Ψ_i, each with finite support of a different width.

- A wide basis function examines a large region of the signal and resolves low frequency details accurately.
- A short basis function examines a small region of the signal to resolve the space details accurately.

4.1.5.4 Tree structured wavelet transform (TSWT)

The TSWT determines channels (subimages) dynamically and decomposes only channels (subimages) with significant energy. It is effective on textures with middle frequency channels. Its decomposition is applied to an output of a filter and it gives a non-redundant representation. The non-subsampled version of the TSWT is called the wavelet frame decomposition. The TSWT provides the space/frequency localisation bases which reconstructs the original signal from wavelet coefficients. Features from channels with small energy are sensitive to white noise. Filters used in the wavelet transform do not change between two consecutive scales.

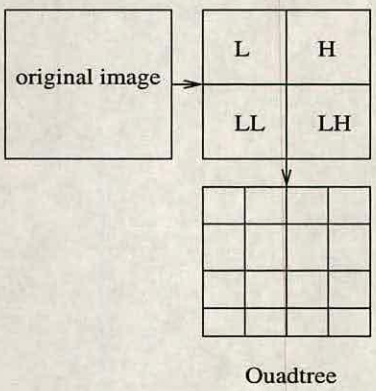


Figure 4.2: *The Quadtree*

The quadtree in Figure 4.2 [79] performs a structured variable block when decomposing an image representing fine and coarse detail. It is generated by the successive division of an image into 4 equal quarters. A test on 4 adjacent sub-blocks is performed to determine whether they should be individually coded or merged. Additionally, it separates high detail and low detail regions in terms of block size. There is saving in computation time as large areas with low frequency are coded at once.

4.1.5.5 Wavelets Summary

The study in this chapter suggests that the wavelet transform is more suited to analysing paper images than the other transforms because of its adaptive window. The problem is that there is an infinite number of bases to choose from when implementing a wavelet. Choosing the right basis is not easy.

4.1.6 Gabor Transform

A Gabor transform [31] is a Gaussian modulated sinusoid in the spatial domain (shifted Gaussian in the frequency domain). Its basis functions are non-orthogonal Gaussian elementary functions, consequently, a Gabor transformed image has some redundancy. This transform is localised in both space (Gaussian window) and frequency. Localisation is a key feature of the Gabor transform, and since texture is a neighbourhood property, it is suited to the optimal extraction of information contained in the texture.

A bank of filters with different orientations are constructed such that they capture a range of tex-

ture type [80]. The motivation for this technique is the human visual system which decomposes an image into several filtered images (processes images in a multiscale way [74]), each with grey level values over a narrow range of frequency and orientation [81]. What makes Gabor filter attractive is their profiles which match the visual receptor field profiles of mammalian eyes (cortex cells respond to different frequencies and orientations). Furthermore, its parameters can be adjusted to elliptical region of spatial frequencies of interest. Gabor filters are suitable for images whose energy is in the low frequency region. However, Gabor filter kernels in (4.11) are fixed and have predetermined frequencies and bandwidths and thus they cannot adapt to local variations of detail smaller than the block (mask) size. The following (4.11) is the relation of a Gabor filter

$$\Phi(x, y, \sigma, u, v) = exp - \left(\frac{(u^2 + v^2)}{2\sigma^2} (x^2 + y^2) \right) expj(ux + vy) \tag{4.11}$$

(x,y) denotes position in the spatial domain, (u,v) are the discrete spatial frequencies, $j = \sqrt{-1}$. The σ is the standard deviation (Gaussian widths) and it determines the degree of overlap of the Gabor elementary functions across the neighbouring pixels so that the truncation of the Gaussian tail produces negligible error. The θ gives the filter directionality. The Gabor transformed image is obtained by convolution of a Gabor filter in (4.11) with an image. In trigonometric terms $u = xcos\theta$, $v = ysin\theta$. u and v are frequencies and θ is the angle of the Gabor filter. x and y in (4.11) are given by:

$$x = xcos\theta + ysin\theta, y = -xsin\theta + ycos\theta \tag{4.12}$$

The difference between the real and imaginary Gabor filter is the 90° phase difference. In addition, for the reason mentioned for other spectral techniques in the previous sections, only the magnitude and not its phase improves texture classification [68]. The shape of the kernel can be altered in (4.11) (u,v), by manipulating the Gaussian widths σ [82] to make a filter become sensitive to texture frequencies [82](u,v), the size of texels and orientation θ. The coefficients become more compact with a decrease in σ whereas the elementary functions become nearly orthogonal as they get closer to the FT basis functions (sines and cosines) whilst the spatial resolution deteriorates and as consequence, texture information is lost. To obtain the right size of the Gaussian width involves a lot of experimentation. In terms of texture, shape and orientation are important. The energy which represents texture of the Gaussian window lies

inside the Gaussian window.

This filter can be designed to treat the points around the pixel of interest of the receptive field unequally [82] (especially where textures differ only in their micro pattern structures). This is accomplished through the use of a combined filter formed from the main Gaussian (unshifted), a left shifted and the right shifted Gaussian [82], then wavelet transformed to produce a set of discriminative filters. But since here we are interested in the surface appearance and not the structure of objects, this type of filter (filter described in this paragraph) is not relevant.

An increase in σ reduces the sensitivity to shorter lines. Parts of features (lines) [80] disappear when their spectrum is larger (higher gabor values) than those of the Gabor filter in a chosen direction. The Gabor filter is sensitive to edges, lines, micropattern size and collinear and elongated patterns at specific orientations [80]. For Gabor filters, small angles result in identical reconstructions. The 2-D Gabor filter distinguishes texture [82] by the frequency and the structure of texels (the primitive for texture) [82]. In terms of application on paper, σ could be used to filter smaller features. The maximum response only occurs when the spectrum of the Gabor filter matches or is greater than that of the line [80]. The Gabor filter decomposes filtered images with limited spectral information [81]. The energy in each subimage is concentrated in both the spatial frequency channel and in space.

The Gabor filter, just like wavelets, can analyse an image progressively from coarser to finer details. This multiscale representation of images is useful as it covers a range of frequencies of occurring textures [81]. Additionally, it exploits its optimality in minimising the joint 2-D uncertainty in space and frequency [81]. Filters with a smaller bandwidth in the spatial domain discriminate well among different textures [81]. The frequencies and orientations that best describe the texture in an image are obtained empirically. An extensive discussion on why Gabor has been chosen ahead of wavelets in this work is included in the chapter summary.

Windowing distorts the spectral estimate due to leakage. A wide window results in the loss of spectral resolution due to an overlap on different regions. Decomposition is localised, consequently, there is a match only with a given periodic signal.

The features computed from the Gabor coefficients comprise the mean, energy, entropy, angular second moment and variance [59]. Texture energy [83] and first order statistics [81] have been computed before in a neighbourhood from the filtered images. The application of the Gabor filters are in texture classification, texture recognition, motion tracking, texture analysis, in

remote sensing applications [84] and edge detection [80].

4.1.7 Summary

The Gabor filter's localisation property is suited to analysing textured surfaces as in chapter 2 texture was defined as a neighbourhood property. Gabor filters compare local as well as global texture information. However, its main handicap is that it has a fixed kernel. Additionally, the optimisation of its parameters, for example, the standard deviation involves a lot of experimentation. Just like wavelets, the Gabor transform is not sensitive to areas of constant amplitude. Its advantage, however, is that it obtains a local orientation estimate even for corrupted images. Gabor is continuous hence it is computationally expensive.

4.1.8 Discrete Cosine transform (DCT)

The discrete cosine transform [85] is a spectral technique. It decomposes an image into sinusoids using its cosine basis functions.

The relation for the DCT is given by:

$$\text{C(u,v)} = \frac{1}{2N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y)[cos(2x+1)u\pi][cos(2y+1)v\pi] \tag{4.13}$$

N is the width and length of the image. x and y are image coordinates. Its implementation involves splitting an image into typically 8×8 blocks. 8×8 is the standard recommended by the JPEG (Joint Photographic Experts Group) [86]. These blocks (subimages) are convolved one after another with a DCT. The results from all the subimages are then combined at the end of the computation.

The resulting DCT transformed image is of the following form:

- The subimage has zero frequency (DC) at the top left corner.
- Frequency progressively increases (AC) towards the bottom right corner of the DCT subimage.

- The highest energy is typically at the top left corner whilst the bottom right corner of the DCT image has the lowest energy.

In terms of texture, the DCT image has good variance distribution, a recipe for texture discrimination. The variance of the DC coefficients is typically larger than that of the AC coefficients [87] thus DC and low frequency components carry more useful information contained in texture in the form of the variance of the convolution mask output [83].

A larger DCT block size results in a diminished influence of neighbourhood pixels as some of them are distant from the current pixel. In addition, a larger block increases computation costs and also violates the assumption of uniform illumination within a neighbourhood. Empirical evidence has shown that any window above the 8×8 pixel size does not improve the DCT result significantly. The DCT technique also suffers from blocking noise (due to the boundaries of the 8×8 subimages) which distorts the result. Additionally, DCT coefficients in adjacent 8×8 blocks are strongly correlated. DCT masks act as spatial bandpass filters in the same way as Gabor filters do.

In comparison with other spectral transforms, the DCT is an optimum transform outside the Karhunen-Loeve Transform (KLT) based on the decorrelation between pixels [88]. The KLT in the form of the principal components analysis is covered in chapter 6 where it is used as a feature selection technique.

The DCT is a linear transform. What makes this transform input independent is that due to the principle of superposition which is associated with linear systems, an input can be recovered from the inverse of this transform. As an illustration, we split an image x into subimages x_1 and x_2 and then filtered them using a filter f resulting in $y_1 = f(x_1)$ and $y_2 = f(x_2)$ filtered subimages. The linear combination of these filtered images should give the same result that will be obtained if the filter were applied to the image x, i.e. $y = f(x)$. The reverse should also hold.

The Karhunen Lo'eve Transform(KLT), not the DCT is the optimal transform in an information packing sense[89]. However, because the KLT is data dependent, obtaining the KLT basis images for each subimage is a nontrivial task. That is why the KLT is rarely used in practice.

Transforming data into another domain in the context of frequency domain, results in statistical independence between pixels. DCT decorrelates pixels, i.e. it converts statistically dependent

pixel values into independent coefficients.

Wavelet transforms achieve a computation performance comparable to DCT whereas they require significantly less computational effort $O(n^2)$ compared to DCT's $O(n^2 logn)$.

The row-column method for implementing the DCT is inefficient in terms of computation as the second DCT can only start after the transposition of the first transform coefficients. The performance of a specific row-column method depends on the realisation of the 1-D DCT algorithm [89]. Whilst the row-column methods of decomposition are slightly less effective than the 2-D direct approach, they have a less complicated procedure for their general implementation on DCT chips.

A competing technique, the DWT only approaches the performance of the DCT in characterising texture with the increasing size of the image and it is less computationally complex. The reason is because the bigger the image or any window size greater than the 8×8 window, contributes very little to the accuracy of the result and also the more expensive it becomes to compute the DCT whereas the DWT since it depends on subsampling, a large area, statistically has more information and thus it is bound to produce a better result.

The applications of the DCT are in video coding [90], in image decomposition and texture representation.

4.1.9 Summary

The DCT is more sensitive to input patterns due to uneven distribution of energy in the frequency domain. It is widely used and has been implemented in many DCT chips. Information from texture in the orthogonal (DCT) masks is represented by the variance of the convolution mask output [83]. The DCT, because of blocking, can result in a low correct classification when used for paper surface quality assessment.

4.1.10 Chapter Summary

Spectral techniques have a potential of characterising the surface appearance of paper. However, because it is not easy to make them become sensitive to the feature sizes of interest, they might not be suitable for paper surface quality assessment.

The window for the WFT has a constant aspect ratio and hence a constant space and frequency resolution. In contrast, the aspect ratio of the window for the DWT changes whilst like the WFT, the window area remains fixed. The sampling rate on a window for DFT and WFT is uniform whereas that of DWT decreases according to Nyquist with an increase in scales. Unlike in the DFT, errors in the DWT do not corrupt the entire transform.

The DWT, Gabor and the WFT yield both frequency information and space information. The DWT is only outperformed by the DFT where stationary sinusoidal signals dominate [73]. The DWT overcomes the DFT frequency localisation problem in the spatial domain. The DWT generates coefficients where transient signals occur, whereas the DFT generates them over the entire span of the image. Wavelets adapt to local variations of a surface profile and are thus suited to the analysis of the surface appearance of paper than the other spectral techniques.

The DFT is least suited to analysing paper images in the machine direction because it considers only the global content of the image. In addition, a defect in the spatial domain, is spread out in the frequency spectrum [73]. However, the DFT can perform better than the WFT on stationary images [73].

The weakness of the Gabor filter and the WFT is their use of a fixed analysis window. Gabor and not wavelets will be implemented in Chapter 7 because its parameters, though empirically obtained , they are easy to optimise. Additionally, Gabor is a continuous transform which should enable it to achieve higher paper classification results. Thus texture and texture analysis techniques can describe the surface appearance of manufactured paper. Additionally, Gabor decomposition has been successful on texture analysis for many years [91] whereas wavelets have only been used extensively for 13 years now and they have not produced significantly better results. Additionally DWT translations are represented in a complex way.

DWT feature is dependent on the part of the image (a feature in one part of the image would look different if it is placed on another position within the same image). This makes texture analysis difficult. Tight wavelet frames do solve this problem. However, these frames are a result of upsampling, consequently, there is a high computational complexity at each stage of decomposition. What makes the wavelet frames different from the other wavelets is that the former is upsampled to achieve the required change of scale, consequently, computational complexity increases with each level due to the increase in the convolution filter length.

Similarly, the DCT, instead of the DFT is implemented in Chapter 7. The DFT was included in

this chapter because despite texture being random, it has some degree of periodicity.

Chapter 5
The Modified Specific perimeter method, Fractal dimension and Lacunarity

5.1 Chapter Introduction

This chapter presents the Specific perimeter method (SPM), the Fractal dimension (FD) and Lacunarity feature extraction techniques. Their potential in analysing the surface appearance of paper will be explored. The results on the optimal values of some paramaters of the SPM are also reported in this chapter.

5.1.1 The Modified Specific Perimeter Method

The specific perimeter method (SPM) [92] is a spatial feature extraction technique based on binary images.

The implementation of the SPM involved locally thresholding an image. An *edgewalker* was used to trace only blobs that satisfied a given aspect ratio and area (bounding box). The edge walker was such that only the pixels that form the boundaries of blobs remained otherwise the rest were set to zero. This area (bounding box) and the aspect ratio ensured that only blobs within a certain size range were captured. The high grey level pixels forming the perimeter (P) of the blobs were then counted. Blobs are isolated regions of high pixel values in a thresholded image. The total number of blob perimeter pixels was then divided by the field of view (A) to give the modified specific perimeter measure.

What makes this SPM a "modified" version is because here local thresholding is used instead of global thresholding used in the traditional SPM. The SPM is calculated as follows:

$$\text{Specific perimeter} = \frac{\sum_{i=1}^{B} P_i}{A} \tag{5.1}$$

where B is the number of boundaries.

Local thresholding is sensitive to local variations on an image and thus it gives a true representation of the surface profile of the paper. However, setting the mask size for this threshold involves a lot of experimentation.

The aim of this section is to assess the power of the SPM technique in discriminating manufactured paper images. The SPM is capable of capturing the properties that are used by humans in discriminating between textural patterns [46]. The key attraction of the SPM is that it measures the graininess of a surface of paper, a feature closely associated with the appearance of a surface profile. A grain is a variation of 1mm or less in the mass distribution of paper. Another strength of the SPM is its use of a single number to describe the spatial distribution of textures.

Most of the work in paper quality assessment using SPM has been in the area of formation. The distribution of fibres in the plane and the thickness of the sheet is called formation [19] (small scale basis weight variations in paper). Jordan et al [92] was the first to use SPM for assessing formation. However, in their work the constraint imposed on the size is that the field of view must be larger than the scale of variation. This thesis has adapted this technique to a feature size of interest proposed by our industrial partners (Tullis Russel & Co.) which is between 0.15mm to 1mm size which in terms of the camera used for capturing the image is equivalent to 20 pixels in size. Roland et al found that an increase in the formation quality led to an increase in the specific perimeter [93]. In like manner, the increase in the surface appearance quality should lead to an increase in the SPM. However, the weakness of the formation measure can be illustrated by a case where light transmission methods covered in chapter 1 are used and the observed blobs might be due to water-marks and as a consequence, a good sample might be disqualified. Thus the analysis of the surface appearance instead of formation is critical. Trepanier et al when measuring formation combined the specific perimeter with contrast [94] and the results were encouraging.

In terms of surface appearance, many small blobs distributed evenly on the surface result in high specific perimeter and uniform or fine paper. Blobs of different sizes which are randomly

distributed on the surface of paper characterise a nonuniform surface of manufactured paper.

A high grey level resolution is useful in sensing feature sizes of interest although it is accompanied by high computation costs. The software that implements the specific perimeter method to evaluate formation quality is available [92].

5.1.1.1 Thresholding for the SPM

The image was local thresholded using the 30×30 mask. The implementation involved moving the mask from pixel to pixel, at each pixel a median value was computed which was then used in turn as a threshold within that neighbourhood (30×30). Thus the threshold was adapted to the local image regions. Here unlike in the case of the global threshold, the benefit is that if there are major changes in the image, vital information is not lost. The size of the mask was determined empirically.

5.1.2 Blob analysis

This technique involves counting the number of blobs generated for the SPM in the previous section. Song et al [50] identified blob defects by means of the Mahalanobis distance discriminant function and then generated statistics from the blobs. Clustered blobs typify a defect.

Danker et al [95] detected blobs using relaxation labelling which uses the probabilistic classification at a given iteration based on the decisions made at a previous iteration. High and low probabilities are initially assigned to pixels based on their grey levels. These probabilities are then iteratively adjusted at each point based on the probabilities at the neighbouring points (high pixel values reinforces high whereas low pixel values reinforces low). Probabilities initially assigned to noise progressively get eliminated with the increase in the number of iterations and as a consequence, high and low probabilities become uniformly high and low respectively. The result is a bimodal histogram and a noise free binary image. This method is appealing but it is computationally expensive and therefore not useful in assessing paper surface appearance for quality.

5.2 Summary

In the machine direction the texture is random, hence any method that uses blobs can discriminate the surface. Large blobs characterise coarse texture and this results in a lower SPM value because the number of high valued grey levels enclosed by a large blob do not contribute towards the perimeter. In contrast, small blobs enclose a small number of high valued grey levels and thus the majority are perimeter pixels and hence a large SPM value is obtained.

The SPM ignores correlations between neighbouring pixels and thus its combination with, for example features from the SGLDM, recoups information on local correlations and eliminates the need for Fourier Transform measurements [92]. The key parameters of the SPM are the aspect ratio and the neighbourhood size and the "minimum" and "maximum" for the bounding box. The bounding box is not a FOV but it is a filter for constraining the allowable sizes of blobs. The aim is to exclude the blobs that are not due to the process that produced the paper, for example blobs resulting from poor handling by the human operator and creases. "Maximum" and "minimum" can be used to differentiate between bounding boxes that have the same area but a different aspect ratio.

The SPM can distinguish fine texture from coarse texture.

Further discussions on the relevance of the SPM to the characterisation of the surface appearance of paper for quality will be discussed with the results from SPM in chapter 7.

5.3 Fractals

Fractals first proposed by Mandelbrot [96] are called irregular segments and in terms of texture fractals are called disordered texture and they are described by surface roughness [97]. The semi-irregular textured structures cannot be modelled by classical geometry and thus fractals are suited to describing them. These structures must be within a certain range of scales on a surface. Since textures are between deterministic and probabilistic, fractals are suited to describing them. What makes Fractals unique is their invariance under changes of magnification and as a consequence, they are "statistically self-similar". Peleg [98] views an image as a "hilly" surface whose height from the normal ground is proportional to the image grey level value. This definition is intuitive.

5.3.1 The Fractal Dimension (FD)

This section presents a scalar called the fractal dimension [99] which is defined as a ratio of the number of features at one scale to the number of features at the next scale. It is a number that quantifies the degree of surface roughness by examining similar structures on an image at different scales (elements being replica of the original structure) hence it is a self-similarity measure. The FD is a mixed space-scale representation with the scaling embedded in the spatial domain methods, for example, the box for estimating the FD is different at different resolution levels. Thus the dimension of a fractal surface contains information about the surface's geometrical properties.

For textured images FD near 2 and 3 means a fine and coarse/rough texture respectively [99]. The FD is scale invariant [100] and also invariant to linear transformations of the data and these attributes are key in the implementation of a vision system. A surface is said to be fractal if the FD is stable over a wide range of scales. The local FD on a window adapts to variations in texture patterns on an image surface.

What makes the study of fractals attractive is that natural images exhibit fractal behaviour over a limited range of scales and that FD uses a single number to characterise texture. Finding a suitable scale is difficult. Additionally, FD's performance is correlated with human judgement of surface roughness [99, 101, 102]. The human visual system (HVS) is a benchmark in this work (does the technique capture what is seen on the surface of paper by a naked eye?). The dropping off exponentially with the distance between points associated with the scale invariance is typical of both the HVS and the FD. In addition the FD is insensitive to noise.

The techniques that compute FD include the Hiaguchi [103], the Katz [104], the Petrosian [105], the differential box counting method (DBC), the absolute intensity difference, the reticular cell counting and the Fourier power spectrum [106]. These algorithms are computationally expensive and application specific. Esteller et al [107] used the Katz to discriminate epileptic states from Intracranial electroencephalogram (IEEG). Hiaguchi's method is more accurate on synthetic data, but is more sensitive to noise. The petrosian has poor reproducibility of dynamic range of synthetic FDs. The Hurst exponent and the fractal Brownian motion (fBm) for estimating the FD are either computationally expensive or have a small dynamic range.

The FD has been widely used in the characterisation of natural surfaces [101].

5.3.2 The Differential Box counting method

This section describes Chaudhuri et al's differential box counting (DBC) method derived from Peleg et al's ϵ approach. This method gives satisfactory results for the 2 to 3 FD range [108].

The implementation of the DBC involves partitioning an image into boxes of size **s**. The **s** is altered progressively to smaller sizes and each time counting **N** the number of boxes covering the surfaces for each size of the box. A least squares fit is then performed using a plot of a graph of **log N** against **log(1/r)** where $r = \frac{s}{M}$ to estimate FD as shown in (5.2). The FD is obtained as a slope of this curve. The first order statistics (FOS) value within a given $s \times s$ box can be used to extract texture information [108]. The features within a box hold the information contained in the texture.

The relations below illustrate the implementation of the FD:

$$FD = \frac{log(N_r)}{log(1/r)}, \qquad N_r = \sum_{i,j} n_r(i,j) \tag{5.2}$$

(5.2) can be expressed in the form y = mx + c which is $logN_r = FDlog(1/r) + C$.

i and j are pixel coordinates in the image.

$$n_{r(i,j)} = l - k + 1 \tag{5.3}$$

The differential part is illustrated by (5.3). k and l boxes receive the minimum and maximum grey level in the $(i,j)^{th}$ grid respectively.

The above relations may be interpreted by considering the probability **P(m,s)** of finding m pixels within a cube (box) of size **s** centred at the current position and the sum of m points within each box gives **P(m,s)**. The number of boxes needed to cover the image is: $N(s) \propto \frac{1}{s^D}$. The knowledge of the FD range, noise level, and window length and the detection of the range in which $N \propto D$ is linear is critical for successful classification.

The DBC has a larger dynamic range, it is faster to compute, it is more accurate and it is effective where the surface structure is self-similar. For many low-resolution images, the box counting (BC) and fBm estimators are unreliable and generate inconsistent results [109]. In BC the range of scale-invariance is assumed to be set by the physical size of the pixel yet the pixel size and the true range are unrelated [109].

The FD defined here has also some disadvantages. The implementation of the FD involves partitioning the image into cubes of size s and as a consequence, it can lead reconstructed images showing blocking, consequently, FD might be unsuitable for paper quality assessment as such distortion could lead to false classification. **N(s)** for computing this roughness measure is calculated without regard to information on the distribution of the pixels within a box or the arrangement and spatial distributions of grey levels and thus it cannot fully describe the texture surface [110] of paper. However, an approach called the multi-fractals [110] in addition evaluates P(m—s) that m points of the set fall in the box of size s. It is computationally expensive though. The FD does not give information on the direction θ in which the texture is degraded. Furthermore, it cannot discriminate two textures that have the same FD but being visually different in appearance. In addition, many FD algorithms saturate before covering the full range of possible values(2 and 3) [111]. The discrimination using FD is limited by the average resolution and algorithms for estimating the slope of the non-linear log-log plots [109]. Keller et al [112] reports that FD alone is not powerful to characterise all types of natural textures totally. A very small or very large region size results in a suboptimal FD value. Kouzani et al [113] solved this problem by computing the FD of regions of different sizes around a pixel and then taking an average. The disadvantage of Voss' method [97] is that it overestimates FDs close to 2 and underestimates the FDs close to 3. Keller modified it by approximating by linear interpolation the surface between neighbouring points [112]. Keller et al [112] combined lacunarity $\Lambda(L)$ with FD managed to discriminate visually distinct texture.

F-matrix proposed by Hiroshi Kaneko [114] is good for both texture analysis and texture classification. The FD is a scaling factor for the image surface area covering number against the scale transform whereas the F-matrix is a scaling operator from the scale space to the 2-D space, whose components are the covering numbers of the mutually orthogonal section curves on the image surface. F-matrix values depend on the choice of the image coordinate system, and consequently, they contain the directional and configurative information of the image, which are lost in the FD. The essential property of the F-matrix is determined by its eigenvalues.

5.3.3 Lacunarity

Lacunarity [97] is a measure of the distribution of gaps on a surface and a fractal is lacunar if gaps are large. Lacunarity captures the second-order statistics of fractal surfaces and also quantifies spatially random texture [115]. Maximum lacunarity occurs when the window size

equals the spatial resolution of the image [115]. This measure is small for dense texture and large for sparse, coarse texture.

The rate of increase of lacunarity $\Lambda(L)$ from 0 to 1 reflects the size of the texture primitives. Results by Kux et al on an image with a sequence of objects at a particular scale, indicate that lacunarity of a spatially random binary process decays slowly until the window size exceeds the scale of the objects and becomes rapid thereafter [115]. Related applications of lacunarity have been on lungs [116]. Lacunarity in combination with the FD has also been used for segmentation using k-means clustering [117] and results were good. This is the relation for lacunarity ($\Lambda(L)$):

$$\Lambda(L) = E\left[\left(\frac{M}{E(M)} - 1\right)^2\right] = \frac{M^2(L) - [M(L)]^2}{M(L)^2} \tag{5.4}$$

where $M(L) = \sum_{m=1}^{N} mp(m, L)$ and $M^2(L) = \sum_{m=1}^{N} m^2 p(m, L)$. M (L)- is the mass of a fractal and E(M) is the expected mass. Lacunarity therefore measures the deviation between the actual mass and the expected mass.

5.3.4 Summary

Feature	Measures	Good Paper Surface
$\Lambda(L)$	gaps	low
FD	coarseness	low
SPM granularity	fineness	high

Table 5.1: *This is a summary of the lacunarity, FD and the Specific perimeter measures.*

P(m,L) is the probability of distribution and it defines the probability that there are m pixels in a box of size L which is centred about a pixel on the image surface. The mass is then the total sum of all these probabilities. FD measures roughness of an $n \times n$ block. The mass of a fractal set is dependent on the length of the measuring yardstick. The mass M is the total number of pixels of the whole fractal image.

5.3.5 Summary

Fractals are related to Wavelets [75] and Gabor filters [31] because they also employ the notion of scale when describing texture. The use of FD in combination with lacunarity have a potential for characterising the surface appearance of paper. This combination has been found to be useful [117]. However, they are unlikely to achieve high performance as it will become more clearer in the Chapter summary that follows. The results on surface appearance of paper from the FD and lacunarity are shown in Figure 7.13 in chapter 7.

5.4 Chapter Summary

The modified SPM in this context is suited to characterising the surface appearance of manufactured paper because the images for this work were captured in the machine direction (MD) (paper has a random pattern in the MD). The SPM excels on such data. However, the SPM ignores correlations between neighbouring pixels and thus its combination with, for example features from the SGLDM will possibly recoup this information and eliminate the need for Fourier Transform measurements [92]. The modified SPM excludes the blobs that are not due to the process that produced the paper, for example blobs due to creases resulting from handling by the human operator.

Natural images exhibit fractal behaviour over a limited range of scales, however, finding a suitable scale to compute FD is difficult. Kouzani et al [113] solved this problem by computing the FD of regions of different sizes around a pixel and then taking an average and this is complex.

The problem of two textures that have the same FD but being different in appearance is solved by using the FD in combination with lacunarity [117]. The blocking problem due to the nature of the implementation of FD can result in false classification. Furthermore, computing **N(s)** for this roughness measure ignores information on the arrangement and spatial distributions of grey levels within a box and thus it cannot fully describe the texture surface [110]. In addition, many FD algorithms saturate before covering the full range of possible values (2 up to 3) [111]. The FD does not have the directionality features θ in which the texture is degraded.

Lacunarity [97] is a measure of the distribution of gaps on a surface and a fractal is lacunar if gaps are large. It quantifies spatially random texture [115]. This measure is small for dense texture and large for sparse, coarse texture.

Related applications of lacunarity have been on lungs [116]. However, lacunarity in combination with the FD has been used for segmentation [117] and results were good thus FD in combination with lacunarity will be used on manufactured paper.

Chapter 6
Feature Selection and classification

6.1 Chapter Introduction

This chapter presents a discussion on a few chosen feature selection strategies.

6.2 Feature selection

Features extracted from images are used as inputs to a classifier. Some of the extracted features might be correlated with each other whereas some might be just useless or noisy when used on the task domain. The main purpose of feature selection is to get only a subset of uncorrelated and highly discriminative features capable of separating data into their respective classes in the feature space.

A feature in a feature space is represented by one point. The reduction in the dimension of the feature space has the added benefit of an increase in the classification speed with an acceptable classification accuracy. Bellman's curse of dimensionality [118] is thus averted.

6.2.1 Introduction

The selection of a subset of highly discriminative features from a set of features is called feature selection. Feature selection can be implemented manually and automatically. Whereas the automatic feature selection approach is faster than the manual approach, however, it can give a suboptimal result. The advantage of the manual approach is that it incorporates intuition (features selected using the knowledge of the problem) and thus in most cases it produces an optimal result. However, the disadvantage is that it is labour intensive. The summary of feature selection techniques is shown in Figure 6.4.

Feature selection in this work refers to automatic feature selection unless stated otherwise. Feature selection strategies normally incorporate a criterion (error) function and a search algorithm.

The criterion functions are problem specific and they measure the correlation between features and the effectiveness of these features in separating classes of data. The search algorithm maximises the chosen criterion function. Examples of criterion functions comprise the Euclidean distance and the classification accuracy.

The sections that follow present a comparative study of a select few feature selection techniques.

6.2.2 Genetic Algorithms

The genetic algorithm (GA) [119] is a non-exhaustive adaptive search technique that evolves a population of better individuals based on natural selection. The population in GA is a collection of chromosomes and each chromosome in a population is a potential solution to the problem. The motivation behind a genetic algorithm [119] is the Darwinian evolutionary theory.

When the GA is used as a feature selection strategy, the population is initialised randomly with some of the training data used as initial chromosomes. A fitness (evolution step) value is then calculated and only the fit chromosomes (highly discriminative features) are reproduced and the rest are rejected. This process repeats while the fitness between the parent and child population are different which in terms of textured paper is minimising and maximising the intra-class and the inter-class distance respectively. The fittest features are maintained and the solution progressively improves through generations. In this context, fitness describes the ability to distinguish between different texture images. Figure 6.1 is a summary of the implementation of a GA. When classification accuracy is used with the genetic algorithm (GA), feature selection becomes very inefficient since a classifier is trained for every genetically manipulated feature subset. One feature on a chromosome is represented by one gene and its presence or absence is denoted by a gene with a value of 1 or 0 respectively. The length of a chromosome is the number of features present.

The GA randomly selects a common point in the selected chromosomes and then exchanges their corresponding bits leading to 2 new individuals. This process of exchanging bits is called *crossover*. The main features of a GA are reproduction, and then crossover which is followed by *mutation* where a bit is altered from 1 to 0 or vice-versa based on a specified probability. The mutations and *selection* result in a stepwise optimisation of features. Mutation prevents the GA getting stuck in a local optimum. However, mutating the most significant bit can result in a weaker individual. The values of crossover and mutation, are typically 0.6 and 0.001

respectively and they are empirically determined and this makes this method suboptimal.

The GAs are recommended for solving complex problems with a large number of features [119]. GAs are relatively sensitive to noise.

A GA is an improvement over random and local search methods. Its features are selected as a unit, and the interaction between different features is tested as a group. GA is computationally efficient when used on a larger number of features. Furthermore, it does not need domain knowledge for optimum classification.

The application of GA has been in condition monitoring amongst others and the features used were statistical, spectral and wavelets [120].

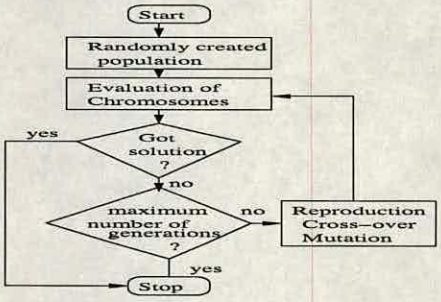


Figure 6.1: *This is a flow diagram for the implementation of a genetic algorithm.*

6.2.3 Backward Selection

The backward selection [121–123] selects the best subset of features. Its implementation involves starting with a full set of features. One feature is held back at a time and then all possible combinations of the features in the remaining set are obtained. These subsets are evaluated individually using a criterion function. The best subset is chosen and the process is repeated on this subset to get the next best subset. At each stage the subset which performs best among all other subsets is selected. The number of possible combinations becomes prohibitively high with an increase in the number of features.

6.2.4 Forward Selection

In contrast to the backward selection [124], the forward selection [121] starts with an empty set. It then evaluates individual features and then the best selected feature is added to the empty

set. This feature once added is no longer available for evaluation for subsequent selections. The process is repeated until all the best features have been chosen.

This technique fails to pick two features that are poor individually, but whose combination give a highly discriminative performance. This happens when these two features are highly correlated and when the second feature is assumed to be giving little extra information for discrimination. This little information might be crucial for the success in separating the different classes of paper.

This technique is less computationally expensive than the backward selection. There is also an improved version, the sequential forward floating selection (SFFS) introduced by Pudil et al [125].

6.2.5 Branch and Bound feature selection strategy

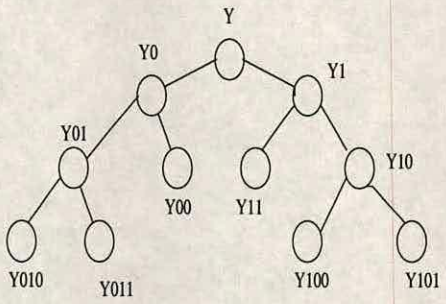


Figure 6.2: *This is a diagram for the branch and bound algorithm. At each node of a tree, there is a branch and this branch terminates once a solution has been found.*

The *Branch and Bound algorithm* [123, 124] generates portions of the solution and computes the criterion for the nodes (shown in Figure 6.2) and in this context, the solutions are subsets of highly discriminative features.

Its implementation involves subdividing an initial search region into subregions ("branching") which are in turn considered by *bounding* an objective function value and then subdividing in the same way as the initial region. The goal is to reject large subsets of non-optimal solutions without recourse to exhaustive enumeration to evaluate them. Whenever a suboptimal partial sequence of nodes satisfies a criteria, the subtree under the node is rejected and enumeration begins on partial sequences which have not yet been explored.

As an example: let $f(x)$ be the function to be minimised, subject to $x \in X$ and where X is a finite set of possible solutions (set of discriminative features). A list L of outstanding (active) feature subsets is kept and the cost U of the best possible solution found.

The recipe for the implementation of this algorithm is as follows:

step 0: Set $U = \infty$. Discard any poor solutions. Treat the remaining solutions as one subset.

Go to step 2:

step 1: Branch step: select one of the remaining subsets and then break it into 2 or more subsets.

step 2: bound step: For each new subset, X, compute $l(X)$

If $l(X) \geq U$, we eliminate X. If $l(X) \leq U$, we reset $U = l(X)$, and X is stored as the best solution so far. The criterion is then re-applied to other subsets until the optimal solution is attained.

In step 1, the Best bound rule which partitions the subset with the lowest bound can be used. It aims for an optimal solution and discards larger subsets. Alternatively, the Newest bound rule which partitions the most recently created subset can be used. Its advantage is that it does not jump around the tree too often, hence it is less computationally expensive.

The interpretation of Figure 6.2 is given here and it should assist in understanding how this algorithm works.

The tree (task) is broken into Y0 and Y1 such that a certain feature is included or not included in the desired feature subset. If neither sub-problem is eliminated by the first tests, then Y0 is broken into Y01 and Y00 such that feature 1 is not included and feature 2 is included or not included in the desired feature subset.

In figure 6.2, Y00 is infeasible and it is thus eliminated.

Similarly, Y1 is broken into Y11 and Y10 which contain feature 1 and not feature 2. It does or does not contain feature 3. Y11 is eliminated using a set criteria.

Y10 is broken into Y101 and Y100 which contain feature 1, and not feature 2 and also not feature 3. Thus Y100 is eliminated.

Y01 is broken into Y011 and Y010. Since $l(Y011) \geq U$, Y001 is eliminated using a set

criteria. As for Y010 it is infeasible, and is eliminated. L is now empty.

The key is getting the conjugate variables of the discarded ones. The threshold value of subsets of features must be fixed in order to obtain an optimal subset of discriminative features. The Branch and Bound algorithm is restricted to small sets and its speed is problem dependent.

6.2.6 The Principal Component Analysis

This section presents a feature reduction method called the principal component analysis (PCA). The PCA derives its basis vectors from the covariance matrix.

6.2.6.1 Covariance

The covariance matrix measures in this context, the linear association between features (data). In a 2-D plot of features, scattered and close points imply a weak and a strong association between data values respectively. If these data points decrease or increase together, the covariance matrix is positive. The negative and positive terms tend to cancel when there is no trend in the data resulting in a covariance of zero. If the components of a random vector are mutually uncorrelated, the covariance matrix is diagonal [30]. Diagonal elements of a covariance matrix are always positive.

The texture features can be expressed in the form of an M^2 dimensional vector x_i as follows:

$$\begin{bmatrix} x_{i1} \\ x_{i2} \\ . \\ . \\ x_{ij} \\ . \\ . \\ x_{iM^2} \end{bmatrix}$$

The covariance of **x** vectors is:

$$\mathbf{C_x} \simeq \frac{1}{\mathbf{M}} \sum_{\mathbf{i=1}}^{\mathbf{M}} (\mathbf{x_i} - \mathbf{m_x})(\mathbf{x_i} - \mathbf{m_x})' \tag{6.1}$$

The mean vector is of dimensionality M^2 and the covariance is an $M^2 \times M^2$ matrix. A covariance shows the spread of samples around their respective class mean. In (6.1) M is the number of features in the input matrix, C is the covariance and m is the mean. Normalising each set converts the covariance between features into a correlation. The zero mean data (data in which the mean has been subtracted) in (6.1), eliminates the average contrast in textured images and it is accomplished through high pass filtering [126]. Average contrast tends to obscure information contained in the data.

6.2.6.2 Principal Components

The aim is to reduce the dimensionality of the inputs to a classifier by reducing the number of features and a related benefit from this is an increase in classification speed albeit a slight reduction in classification accuracy in some cases. Although the PCA in this work was employed in feature selection, it is felt that its versatility is explained better by describing how it functions on images and texture.

The implementation of the PCA involves creating a covariance matrix from the features extracted from images and then diagonalising this matrix to produce the PCA's basis vectors, the eigenvectors and the eigenvalues. An eigenvalue is the variance of the data (feature) values [127] about a hyper-plane defined by an eigenvector. The eigenvalues are arranged in descending order such that the corresponding eigenvectors are derived in a decreasing order of importance. The largest eigenvalue carries the largest variability and this variance is key to correct texture classification. The eigenvectors also called the principal components represent the direction of greatest uncertainty, hence information along the direction in which the inputs exhibit most variation is retained. Eigenvectors are therefore adapted to the underlying information. Additionally, eigenvectors of a normal matrix are orthogonal and this property results in decorrelated information from features and consequently there is minimal or no redundancy in the resulting output. The data is projected onto the chosen most significant eigenvectors as indicated in Figure 6.3 and the result is a linear combination of the original features defined by the eigenvectors [128]. This eigenfiltered data carries discriminative information and it is used as the classifier input.

In terms of texture, the texture with a high directionality is characterised by one or more high eigenvalues with eigenvectors having strong directionality in the corresponding direction. Eigenvectors are thus adapted to the texture studied [129]. The first principal component corres-

ponds to the largest eigenvalue and it is the average [127] of features and it contains in excess of 80% of the variance of the original feature set. Common characteristics of these features appear in the first components, while differences and residual information are in subsequent components [130]. A non-uniform variance on all components suggests a potential for dimensionality reduction [131]. Using a few most significant eigenvectors results in the data with a new variance and energy has been compacted into a few coefficients.

Ade et al found that the energy computed at the output of a bank of eigen-filters is useful in texture classification [127]. Comparative studies [7] show that the PCA combines the attributes from the co-occurrence and the laws approach [132]. Laws' approach is a feature extraction technique that uses local texture energy measures. It extracts information from small image windows just like the human visual system. The analogue of the PCA to laws' detection of edges property is the PCA's eigenfilters. The analogue of the PCA for co-occurrence matrices' grey level transition property is the $(m - \bar{m})$ term of the covariance matrix. PCA is recommended for data with a multivariate normal distribution.

PCA fails to discriminate between variance due to data and that due to noise [128]. Additionally, eigenfilters comprise also gradient filters and thus they are susceptible to noise.

6.2.7 Summary

The PCA is a linear transformation of the original features and the result of this transform is an uncorrelated feature set which maximises the variance contained in the original feature set [133].

The coefficients of the PCA are defined if two or more eigenvalues are not identical. The PCA focuses on the variance of the inputs which may ignore its correlation to the output and this is a weakness.

The PCA is robust to noise, it is simpler and computationally inexpensive. In terms of the characterisation of manufactured paper, the PCA as a linear technique is not suitable because this data (paper samples) is non-linear. The conclusion that the data is non-linear was based on visual assessment, thus it was crucial to confirm this by using a linear technique such as the PCA.

This algorithm is illustrated by Figure 6.3. The size of eigenvalues is n^2 where n is the size of

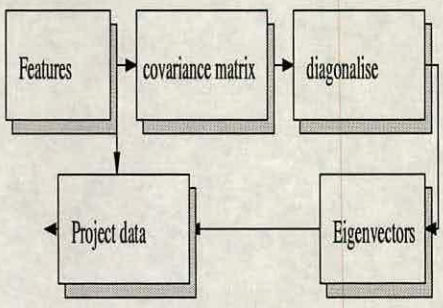


Figure 6.3: *This diagram shows stages for computing the principal component analysis. The inputs in this context were features*

the input feature set. The product of each eigenvector with the features results in n^2 principal component data.

6.2.8 Manual Selection

The manual approach is a feature selection strategy that is based on the investigator physically picking features using the knowledge of the problem and the information that those features characterise. This feature selection involves picking features by hand. This incorporation of intuition although it is labour intensive it is key to the success of this work.

The manual feature selection technique was implemented in conjunction with classification acting as the error criterion. A large subset of features thought to be relevant to the problem were used as inputs to the classifier. The classifier was trained and tested using selected architectures (number of hidden units) and the generalisation performance (classification) noted. Features that were viewed to be less important were removed one at a time and the network was then retrained. The generalisation performance was noted after the removal of each feature. If the removal of a feature resulted in a significant fall in generalisation performance then that feature was put back in the set of selected features. This approach becomes computationally expensive for a larger set of features, for example, dealing with 18 features and say architectures of, 5, 10, 15, 20,25 and 30 hidden units and for each set of hidden units making 10 classification runs.

The manual feature selection strategy's incorporation of intuition makes it attractive for this work.

6.2.9 Feature Selection Summary

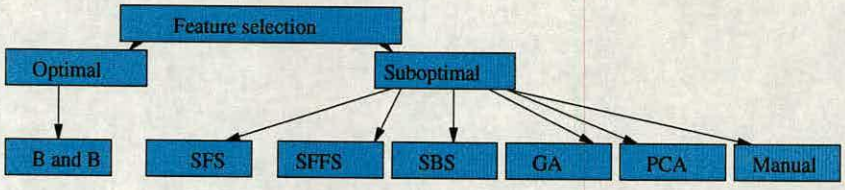


Figure 6.4: *This is a Graph showing feature selection techniques. Manual is the manual selection , B and B is the branch-and-bound, GA is the genetic algorithm, SFS is sequential forward selection, SBS is sequential backward selection, SFFS is sequential floating forward selection.*

The block diagram Figure 6.4 is a summary of the techniques that were covered in this section. The features selected by these strategies are used in this context, as inputs to the MLP classifier.

The GA evolves a population of better individuals in this context highly discriminative features. This evolution is based on natural selection. However, the search space explored by the GA increases quickly and its performance degrades with the increase in the size of the feature set. Additionally, the values of GA parameters are empirically obtained. When a classifier is used as a fitness function for the GA, then the classifier has to be trained for every genetically manipulated feature subset. Thus this procedure is cumbersome.

In the Forward selection, a selected feature is no longer available for further selection, consequently, it misses out on two features which perform poorly when acting individually, but which are highly discriminative when combined. Additionally, several networks have to be trained in order to select an optimum architecture when used with neural networks.

The backward selection starts with a full set of features. One feature is held back at a time and then all possible subsets in the remaining set are evaluated individually and the subset which performs best is selected. The number of possible combinations becomes *prohibitively* high.

The Branch and Bound algorithm selects features through the subdivision of the feature set that involves branching and bounding of an objective function value. This process is repeated cyclically until an optimal feature set has been found. This strategy is computationally more expensive than the PCA. Furthermore, the Branch and Bound algorithm is restricted to small sets and its speed is problem dependent.

Feature selection using the PCA does not give information about the quality of the features.

Additionally, the PCA computes all the features before the transformation matrix can be applied, hence it incurs a higher computing power overhead. However, this is offset by the lower computation power required on a classifier. In terms of comparison with the GA, the PCA is almost 20 times less computationally expensive than the GA. Furthermore, the GA will need to search the space every time a classification is performed.

The manual selection of features is labour intensive. The solution from the manual approach has a scientific explanation as incorporated in this approach is the prior knowledge of the problem by the investigator and the knowledge of the features that are relevant to the problem. There is however, a speed disadvantage during the classifier training phase compared to the automatic approach. Additionally, there could be concern about relying too much on the human expert knowledge, however, once the optimal classifier has been built, addition of a feature in the future does not necessitate having to go through another long selection procedure.

In comparison the manual feature selection can take from a few hours to months whereas the PCA can be completed within a few hours to a few days.

The manual feature selection approach in conjunction with classification accuracy will be adopted for this work. This strategy is justifiable over the other feature selection strategies because the classifier acts as an error function that is interacting with the human expert (the author of this thesis) in order to come up with an optimal solution.

6.3 Classification

Classification is the assigning of a sample to one of the given categories. The available classification algorithms are many and varied. Some classifiers need an input-target pair (supervised classifiers) and the other type needs only inputs (unsupervised approach). This section presents a discussion on neural network classification.

6.3.1 Introduction

The aim is to get a classifier that is capable of separating paper into three distinct classes the "poor", the "average" and "good". The choice of 3 classes came from our discussion with the industrial partner as it had become difficult for the human expert (me and industrial partner) to visually split the data into more than 3 classes. The inputs for this classifier must be features

extracted from texture. A classifier supplied with good features must be capable of separating the given paper images into the desired categories.

The neural and statistical [134] classifiers are widely used. The latter are grouped into parametric and non-parametric. The parametric form assumes a Gaussian distribution of data whilst the non-parametric makes no such assumption. The key parameters for estimating a distribution are the mean vector and the covariance. Parametric algorithms in most of the cases perform better than the non-parametric form even if the assumed class distribution is invalid. In the non-parametric form all the samples must be stored and this is a disadvantage. Additionally, the non-parametric form requires a large number of samples to achieve good estimates and hence it is computationally expensive. The parametric form is therefore preferred to the non-parametric form.

The neural classification strategy was adopted over the statistical classifiers in this work because it deals directly with information (it is data-driven). Neural classifiers are known to perform well on non-linear data. A comparison of different neural classifiers will not be done in this work because the focus is on the use of texture in classification and not finding the best classifier.

6.3.2 Neural Networks

In this chapter the Neural networks is discussed. We are not claiming any novelty in the neural networks as standard algorithms and methods have been used. Neural networks (NN) are not model based but are data-driven. They are modelled along the biological neuron. These networks compute any desired function that captures the underlying structure of the data which formulates a model of the system that generated it. This function is represented by the stored (**frozen**) weights obtained during neural network training.

The input data to a neural network can be an image(s), features or grey level values of images. The NN uses a large training set and a suitable network architecture and a training algorithm to adapt the network parameters (weights). Some training strategies involve implementing several architectures to gain confidence in the network's ability to extract relevant information with the architecture that gives the best solution being selected [134].

The neural network (NN) represents non-linear mappings from several input to several output variables. The *patterns* (input features or input-output pair) in a NN are represented by points in a decision space which is made statistical by the variability within and between the classes.

These decision spaces mapped out by the NN are used for assigning inputs it has not seen before but that are typical of patterns it has seen in the past. This process is called *generalisation*. Thus neural networks have a potential of handling novel environments and performing better than statistical classifiers.

Neural networks are recommended where the input is high dimensional, discrete and real valued (there is no restriction imposed on inputs) and where the training time is not critical. They are useful when it is hard to figure out what is going on and when the problem is too complex to be solved by traditional techniques.

Neural networks have also disadvantages. They incur high computational costs especially where large inputs and outputs are involved. In addition, there is always noise in the data because real world data is probabilistic and also due to the human error in data collection (incomplete data) and labelling. Noise tends to degrade the performance of a classifier. Furthermore, the performance of the NN suffers where there is insufficient input data because NN is a statistical method. How much data is "sufficient" for training the network is not known. Moreover, in using an early stopping criterion during training, training can stop before the network gets to a global minimum hence information contained in the resulting "frozen" weights is suboptimal.

In summary, neural networks do not explain the computational process by which the network makes a decision. Training many networks and then averaging can improve the results. The key notion in neural networks is learning and not programming. Neural networks are categorised under learning strategies, namely unsupervised and supervised learning.

6.3.3 Learning strategies

Unsupervised learning [134] is used on unlabelled data and it automatically groups similar input samples into distinct clusters. This approach is thus biologically plausible and it thrives where there is redundancy in the input data. Each cluster represents a distinct class. The desired outcome is a small intra-cluster distance and a large inter-cluster distance between the different classes of data. The clustering criterion is the minimisation of the sum of squared errors (sse). However, the correct number of clusters is not always known and where there is only one class, clustering will try to create two. Additionally, this network has no feedback with regard to outputs. This makes this learning approach suboptimal.

Supervised learning needs an input-target pair in order to learn. In terms of paper classification, a sample image is assigned to one of the known classes.

6.3.4 Linear Classification

The perceptron [134] is a McCulloch-Pitts [118] model of a neuron. It is a 2 class classifier that linearly separates data by using a hyper-plane. It achieves this by the adjustment (tuning) of a weight vector and bias of the neuron otherwise the perceptron learning rule does not converge (decision boundary oscillates between a number of positions). The bias weight allows flexibility over the position of the hyper-plane by shifting the decision boundary in the feature space. Thus the perceptron learning is finding suitable weights "W" such that the output $y = 1$ for one class and $y = 0$ for the other class of data. However, the projection of data onto only one dimension leads to a loss of information as classes which were separable on a larger dimension might overlap one another.

The neurons were implemented in software. Thus in brief, the architecture of a neuron includes the neuron and the linear synaptic links which sum their respective input signals. The output of the neuron is 1 when this sum exceeds a given threshold otherwise it is zero. The neuron is said to fire when the sum from input weights is greater than the given threshold (activation function's output).

The bias is another component of the NN whose function is to position the hyperplane at a point where the hyperplane maximally separates the classes of data. The key function of neurons is the construction of internal representations of their environment.

Whereas the data (manufactured paper) is non-linear, the linear classifier will be used in the classification of this data only to confirm this assertion. Additionally, this is to make sure that we do not use an expensive non-linear method where a fast a linear classification method could have performed equally well.

6.3.5 The multilayer perceptron

The multilayer perceptron (MLP)[134] in Figure 6.5 is a generalisation of a single layer perceptron discussed in the previous section. It has two or more layers of hidden units and it is robust to noise and has good generalisation ability. It is a supervised classification strategy

and thus it guarantees correct classification. The usefulness of an MLP in paper classification is from its nonlinear nature. The MLP used in this work was an off the shelf neural network implementation.

The goal is a trained network that can interpolate between training examples and generalise to the test data. A large number of training samples results in less freedom of interpolation and as a consequence, a phenomenon called overfitting (the network begins to learn noise) occurs and hence generalisation might be poor. This problem is solved by keeping the size of training data to moderate.

The advantage of the MLP is that it has distributed parallel nodes and a failure of a node leads to a slight or no degradation in performance. Such networks are said to be fault tolerant. The

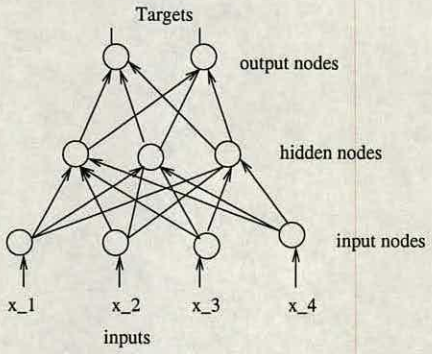


Figure 6.5: *This is a schematic diagram of a multi-layer perceptron*

proportion of misclassified samples is used as a performance criterion.

The MLP's basis are not determined by the Euclidean distance thus normalisation is only performed because of the range of the logistic function (especially at the points where its gradient is zero). A logistic function is "S" -shaped. At the top and bottom part of the "S" shape, the gradient is zero. The benefit of the normalised vectors is also in reducing time taken by the backpropagation when training the MLP.

The bad news is that the MLP is incapable of retraining itself (is not evolvable) for future applications. Thus when a new texture is added to a trained network a new set of "frozen" weights that captures the new underlying function that relates the inputs to the outputs must be obtained through retraining. A 2 - layer MLP has been found to be sufficient for many applications and it is also recommended for this work. Training a classifier involves training,

validating and testing the network as shown in Figure 6.6. More about this will be said in the next chapter.

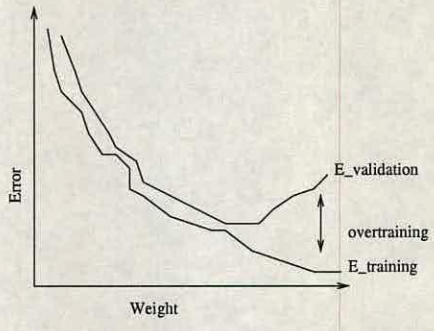


Figure 6.6: *This graphical plot illustrates supervised neural network training*

6.3.6 Training

Training is the tuning of weights in order to produce the desired classification result. Training can be performed in either the batch or the sequential mode. In the latter a sequence of forward and backward computations are performed on a pattern by pattern basis, consequently, the search in weight space is stochastic and the backpropagation is likely to perform well. Furthermore, it takes advantage of redundancy of the training data. An epoch is one complete presentation of the entire training set during the learning process.

In the Batch mode learning is maintained on an epoch-by-epoch basis until the mse over the entire training set converges. A local minimum value is guaranteed in the batch mode and it also provides an accurate estimate of the gradient vector. It is also easier to know what is going on and to parallelise than in the sequential mode. However, the batch mode fails if the input data keeps on changing. Additionally, it requires more local storage for each weight than the sequential mode. The following are the relations for batch and sequential modes respectively:

$$w < -w + n\sum(t_d - O_d)x_d, \qquad w < -w + n(t_d - O_d)x. \tag{6.2}$$

x is a pattern and x_d is a vector, w is a weight, t_d and O_d are the target and actual outputs respectively. In summary, training more than once and training long enough using a few hidden units and then getting average performance minimises errors. If the training error is consistently high, the neural network might not be suitable for the problem or there might not be a

relationship between the inputs and the output or a wrong set of features might have been used. The strength of the neural networks as a data-driven strategy lies in their ability to analyse and recognise complex patterns in the data.

6.3.6.1 The Logistic function

The sigmoid in a neural network squashes or constrains the output range of a signal (the sum of inputs from hidden units in preceding layers) to a finite value. The goal is to ensure that the network outputs represent a posteriori probabilities [135] (the expected decisions).

$$\text{sig} = \frac{1}{1 + exp(-a)} \tag{6.3}$$

where sig is a sigmod. The sigmoidal is biologically plausible because it mimics real neurons. Large initialisation weight values pushes the sigmoid units to the limits (close to saturated or tail ends) and here the derivative is nearly zero and learning becomes painfully slow or stalls.

6.3.7 Optimisation

Optimisation in neural networks involves devising a strategy for minimising the error between the desired and the actual output of the network. There is always one minimum for the linear classifier case. In contrast, the MLP is a non-linear classifier with many minima and its local minimum is not necessarily the lowest point ("A" in Figure 6.7) on the error surface and its global minima ("B" in Figure 6.7) is lower than any point on the error surface.

The goal during training is a global minimum attained through forcing hidden units to adapt to the task domain. This process is called learning and it is achieved through the use of an error function like the Euclidean and an optimisation technique like the line search method, the conjugate method and the gradient descent (methods that go down hill on the error surface) [118]. The aim is to minimise the error "E".

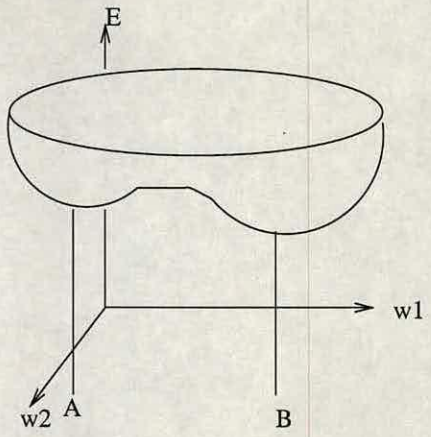


Figure 6.7: *The Graph shows the local minimum point A and the global minimum point B in an Error(E) - Weight(W) space.*

6.3.7.1 Gradient descent

The gradient descent method is a network training algorithm and it is normally used with the backpropagation algorithm.

$$E(w + \delta w) \simeq E(w) + \sum_{i=1}^{W} \delta w_i \frac{\partial E}{\partial w} \tag{6.4}$$

The main objective of optimisation is reducing the ∂E term to zero. where g are all partial derivatives computed:

$$\begin{pmatrix} \frac{\partial E}{\partial w_1} \\ \vdots \\ \vdots \\ \vdots \\ \frac{\partial E}{\partial w} \end{pmatrix} -$$

Therefore $E(w + \delta w) = E(w) + g.\delta w$

$$\frac{\partial E}{\partial w} = o_a - o_d \tag{6.5}$$

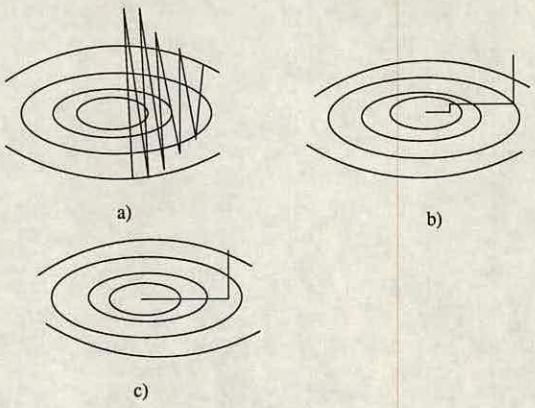


Figure 6.8: *The Graphs show a) the gradient descent, b) the line search c) the conjugate gradient method. The ellipse are error surfaces. The global minima is in th centre(smaller circle)*

where $o_a and o_d$ are the actual and the desired outs respectively. When computing the gradient direction step (local negative gradient of the error function), locally the function is modelled as a plane. This is a search for a minimum of some function. A criteria for stopping optimisation is achieved through the computation of $g = \frac{\partial E}{\partial w}$ in the error - weight space. The weights ($w \Leftarrow w - \eta g$) must change towards the steepest descent and in the direction of the global minima. η is the learning rate.

Finding where we are in terms of local and global minima needs knowledge of the whole space. The gradient descent method is incapable of finding that. This is its main weakness. However, from the gradient information, each evaluation of ∂E brings W items of information. The key is how the slope is falling and how far "down hill" it will have moved before it gets to a global minimum of some function. The desired optimisation strategy must get to a minimum of the error function in the error-weight space in a few steps.

The batch version of gradient descent starts with an initial guess for the weight vector which is iteratively updated such that at each step there is a gradual change in the direction of greatest rate of decrease of error as shown in Figure 6.8 (a). The gradient information improves the speed with which the minimum of the error function can be located and without it we are guessing. The step size that is used in moving down hill is fixed by the learning rate η. A small η results in a small change to the weights in the network from one iteration to the next in the direction of the solution, consequently, it might take longer to reach a minimum. In the case where a big (stepping far away) step size is used, the speed increases and the network might

overshoot the global minimum and result in a poor approximation as shown in Figure 6.8. Thus the learning rate η only scales the pattern vectors (training examples).

A parameter called momentum, reduces high-frequency variations (zig zag movements in Figure 6.8 towards a solution) in the weight space [136]. Large and small weight changes result in large and small momentum respectively. Large momentum accelerates convergence and keeps the network from being trapped in a local energy minima. The momentum term and the η must progressively decrease as the mean squared error (mse) approaches a minimum if rapid early learning with stability are to be achieved.

Poor internal representation within hidden units could result in a local minimum which occurs when different classes of data fall into one class. Hidden units control the flexibility of the decision boundaries and thus if they are increased they can potentially eliminate the local minima. The addition of noise can eliminate a local minima through the perturbation of the gradient descent algorithm. However, the gradient descent algorithm is suited to small tasks and hence this is a drawback. If the error does not decrease over a large number of iterations during training, training must be restarted using new random weights [136].

6.3.7.2 Line search

In the line search method [134] a point is chosen and the error minimised in a direction along a line until a minimum is reached. A new direction perpendicular to the former direction is chosen and followed until a new minimum is reached. The process is repeated until the solution is reached. The line search method unlike the gradient descent is more stable as shown in Figure 6.8. Additionally, it does not use the momentum and the learning rate parameter. Instead of a single step or guessing what the learning rate is, taking small steps is recommended.

6.3.7.3 A quadratic function

Taylor expansion of:

$$E(w + \delta w) \simeq \frac{E(w) + g.\delta w + 1}{\delta w^T H \delta w} \tag{6.6}$$

If H (Hessian) is positive, this models the error surface as a quadratic bowl. A quadratic function can be minimised directly but this requires knowing / computing H, which has size $O(W^2)$ and

inverting H is no mean task. Thus the second order information is there but it is not going to be easy to use it.

6.3.7.4 Conjugate gradient method

An alternative search strategy involves the construction of a sequence of successive search directions such that each direction is conjugate to all previous directions [134]. This approach is called the conjugate gradient and it determines the step in each direction automatically as shown in Figure 6.8 c). Furthermore, it combines the good attributes of the second order (historical information about where it has been) and that of the line search method. The conjugate gradient is widely used and is more efficient, faster and easy to know what is going on. There is no need for computing the Hessian at all with this method. This is a big advantage.

6.3.7.5 Back propagation

The backpropagation algorithm [118] is used for training an MLP. It consists of the forward and the backward pass. It penalises those weights that produced a wrong output. The network is initialised by setting random weights and sigmoids. In the forward pass results from the computation at each neuron from each layer pass through the network layer by layer resulting in the actual output. There is no adaptation of weights in the forward pass. In the backward pass, the actual output is subtracted from the desired output. If there is no error, weights are not changed, otherwise the error is propagated from the output in the direction of the inputs and at each layer computing the local gradient for each neuron and simultaneously adjusting weights. The training set is repeatedly presented to the network until the Euclidean norm of the gradient reaches a certain threshold or when the mse attains a constant value as in Figure 6.6. The weights are then saved (frozen). These weights contain information pertaining to the underlying structure between inputs and the corresponding targets. This knowledge (saved weights) is used by the classifier on test data (data that has not been seen before). The test data is presented using just the feed forward calculations.

6.3.7.6 Classification Summary

Classification is the assigning of a sample to one of the given categories. The supervised classifiers that will be used on paper are the linear classifier and the MLP classifier. The neural clas-

sification strategy was adopted over the statistical classifiers in this work because the former is is data-driven and that they can compute any desired function that captures the underlying structure of the data which formulates a model of the system that generated it. A comparison of different neural classifiers is beyond the scope of this work.

The decision spaces mapped out by the NN are used for assigning inputs it has not seen before but that are typical of patterns it has seen in the past. This process is called *generalisation* and it is what makes the NN attractive.

Neural networks are recommended where the input is high dimensional, discrete and real valued and where the training time is not critical.

NN incur high computational costs especially where large inputs and outputs are involved. In addition, there is always noise in the data because real world data is probabilistic. Noise degrades the performance of a classifier. Furthermore, the performance of the NN suffers where there is insufficient input data because NN is a statistical method. Neural networks do not explain the computational process by which the network makes a decision. The key notion in neural networks is learning and not programming. Training many networks and then averaging can improve the results. The key notion in neural networks is learning and not programming.

Three optimisation techniques discussed in this work and their function is to reduce the difference between the desired output and the actual ouput from a classifier. The gradient descent method recalculates error at each step which is inefficient. This algorithm is suited to small tasks and hence this is a drawback.

The line search method is computationally less expensive and more stable than the gradient descent method. Additionally, it does not use the momentum and the learning rate parameters which are used by the gradient descent method.

The conjugate method constructs a sequence of successive search directions that are conjugate to all previous directions. It determines the step in each direction automatically. It combines the good attributes of the second order (historical information about where it has been) and that of the line search method. It is more efficient, faster and easy to know what is going on. The gradient descent method has been used here instead of the optimal conjugate method because the former was readily available already coupled to the MLP.

What can go wrong in neural network training include memorising the data, a phenomenon as-

sociated with using too many hidden units during neural network training. The network begins to model noise if it is trained well beyond the stopping criteria and this is called overtraining and it is illustrated in Figure 6.6. When ambiguous data (similar input vectors given different labels) is used the NN learns the average of the ambiguous target values which leads to a wrong solution. The data set in which one of the classes has for example, twice more samples than the other classes, is called an unbalanced set. In this set the test samples will always give a wrong result. Extensive work has been done in chapter 7 to rectify this problem.

The MLP neural network with its backpropagation training algorithm are suitable for classifying paper. The weakness of the MLP is in the use of the mse (mean-squared-error) which is accompanied by longer learning times and also the change in mse can occur before reaching the global minimum and this makes the MLP solutions suboptimal. It is therefore profitable to have a classifier that incorporates a cost and at the same time optimising learning. Chapter 7 presents a report on experimentation carried out using the MLP classifier and a linear classifier and also includes a detailed discussion on computing costs using a loss matrix shown in Table 7.8.

Chapter 7
Experimental

7.1 Experiments

This Chapter presents the procedure followed when capturing images from manufactured paper samples. The procedure followed in pre-classifying paper samples is also presented. The experiments and results from preprocessing and classification are covered in this chapter. Suggestions of techniques that might be useful in improving classification performance on this data are proposed.

7.1.1 Training of the human expert at Tullis Russel & Co.

I (author of this thesis) was trained in visually classifying paper samples at Tullis Russel & Co. (our industrial partner). The training was performed once a week over a period of 6 weeks. The expert at Tullis Russsel & Co is the company paper quality controller who has many years of experience in inspecting manufactured paper.

The training was split into two stages. The first stage involved identifying the paper direction in terms of machine direction (MD) and cross direction (CD) and classifying paper samples into 3 classes. The MD is the direction of the conveyor belt system (paper reel) whilst the CD is the direction that is at right angles to the motion of the paper as it comes out of production. The key identity is that the former has a random pattern whilst the latter has a periodic pattern. I managed to master this during the 20 minute training session.

Visual classification of the samples was carried out on different days, in the mornings, in the afternoons and in the evenings. The differences in classification for these different times of the day were not statistically significant. Classification in the mornings and late in the evenings was in the range of 97% to 100%. In the mornings one is still fresh and in the evenings there was no distraction. In the afternoons classification was in the ranges 95% to 100%. The lower classification was attributed to a distraction due to movements by colleagues in the laboratory.

The meetings with the expert at Tullis Russel were held at the times that were convenient to him. Fortunately, these tended to be different times of the day and thus we could measure the variability in classification. At these meetings he would classify the samples. His classification tended to be consistent. The time of the day had no effect on his performance. The reason could be that he does not do many jobs in between and also many years of experience on this job is another factor. It turned out that our results were comparable. However, it will be useful in future to compare the performance of this system with the performance of a range of human experts (from the untrained to the well trained or from the inexperienced to the well experienced).

The following is the recipe used during the training sessions at Tullis Russel & Co to identify MD and CD of the paper:

- place a paper sample on a table that slants at 45° to the horizontal plane.
- switch on the lights shown in Figure 7.1.
- note if the direction is the Machine direction (MD) or Cross direction (CD) on the paper surface.
- rotate the paper sample such that the new position is at right angles to the previous position and note whether it is MD or CD.

The next phase involved classifying the paper samples based on the surface appearance into 3 classes. The recipe that was used during the training session was as follows:

- Put a sample of paper on a table that slants at 45° to the horizontal plane.
- Switch on the lights shown in Figure 7.1.
- Perform blind classification of the sample by comparing it with the 3 specimen paper samples, representing the 3 classes.
- Place the sample in the class that closely resembles it.
- Repeat the process until all the samples have been classified.
- Take the results to the experts at Tullis for comparison.

The key feature in visual classification was the distribution of grains that form the surface profile of paper. Angular illumination was necessary as it enhanced the features on the paper (hills and valleys). 50 samples pre-graded by a team of experts at Tullis Russel were used during training (visual classification).

Initially the correlation between my classification and that of experts at Tullis Russel was 80% and by the end of the course it was 99%.

Once I had mastered the art of visually classifying paper samples, I was furnished with 85 samples that had not been pre-graded. These samples were finally used for the project. They were graded and re-graded up to 5 times over intervals of 4 days spread over a period of 3 weeks. Once I was satisfied and also after demonstrating to my project team at Edinburgh who were not experts in grading paper, the samples were taken to experts at Tullis Russel & Co. and there was 98% correlation with my classification.

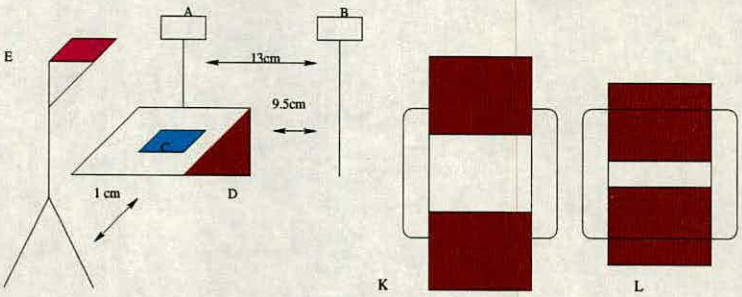
The difference in classification results between the first 50 samples and the 85 used for the project, is that the classes of data in the former were not as overlapping as in the latter.

7.1.2 The optimisation of the Image capture parameters

The key parameters in image capture include the camera's resolution, the illumination received by the sample, the field of view (FOV), the camera's distance and angle to the sample.

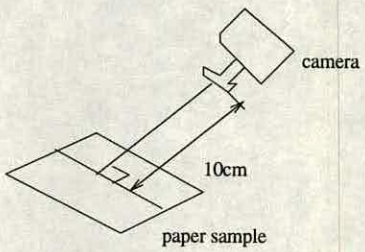
The sample of paper was illuminated at a glancing angle and this approach is not new [137]. The part of the paper sample near the lights shown in Figure 7.1 received more illumination compared with the rest of the paper sample resulting in an illumination gradient. The desired effect was a paper surface with enhanced "valleys" and "hills".

The paper images were captured in TIF format using a Olympus C - 2500L digital camera under the same conditions of light. The resolution of the camera was 1712×1368, consequently detail captured approaches that seen using a naked eye. The resolution of the human eye is $100\mu m$ [92] and the resolution of the captured image is $70\mu m$. This camera has a delayed exposure and thus the automatic mode which avoids vibration due to manual operation was used. A white paper with text was used to set focus. The digital camera angle was set at 90° to the surface of paper as illustrated in Figure 7.1(c), the height of the camera to the sample was fixed at 10 cm, the surface of the table had a slant of 45° to the horizontal plane and the FOV (sample size)



(a) This is the schematic for capturing the image. It is not to scale. A and B are light sources. C is the paper sample. D is the table that slants at 45° to the horizontal. E is the camera resting on a tripod.

(b) This is a schematic for the light source. In K the orifice is 100% open. In L the orifice (is 50% open) is 3cm wide



(c) This schematic shows a camera held at right angles to the sample.

Figure 7.1: *This is the image capture set up used in this work.*

was set to 12.5cm × 10 cm. The distance of the tripod to the table was fixed at 1 cm. The two "Landsco" white light sources were set at 7 cm above the table on the higher end as shown in Figure 7.1(a). The distance between the two light sources was 13 cm. The distance of each mast (holding the light sources) from the table was 9.5 cm.

The results for the optimisation of the light emitting surface area (orifice) in Table 7.1) were obtained from adjusting two shutters (shown in Figure 7.1(b)). A feature was selected on the sample. Then the chosen orifice size was such that when an image is captured the same feature seen on the sample is seen on the image. Secondly, the image had to show minimal gradient illumination which meant a balance having been struck between saturation (due to too much illumination) and a poor signal to noise ratio (SNR) which meant insufficient illumination on the paper. The optimal settings for the orifice were 3cm × 7cm which corresponds to 50% in Table 7.1.

Orifice	image quality
100%	poor
75%	satisfactory
50%	good
25%	satisfactory

Table 7.1: *This is a Table for the optimisation of the size of the orifice shown in Figure 7.1(b) that allows light to fall on the paper (illumination).*

The position of the paper sample and the focus point were marked on the table. The sample was enclosed in a black non-reflecting box in order to screen external illumination and avoid light reflections from the box onto the paper surface. The images were captured a few centimetres from the edge of the paper sample to avoid incorporating defects due to handling. The captured area is supposed to represent the overall quality of the paper surface. The captured images were showing gradient illumination which in image processing is viewed as noise and it must be removed in order to get a crisp image.

7.1.2.1 Database

The experimental database was not collected so that the poor, the average and the good classes of data could be equal, but it was such that it was representative of the paper that is produced in the plant. The data collection was done by our industrial partner (Tullis Russel & Co).

The images (1712 × 1368) from the previous section were cropped into 256 × 256 sub-images

as shown in Figure 7.2.

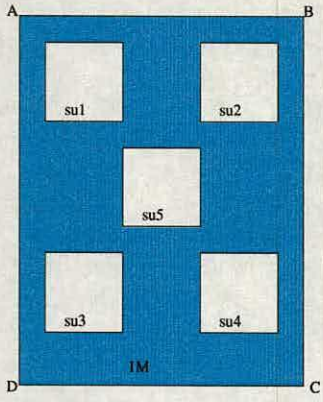


Figure 7.2: *This diagram shows an image "IM" of* 1712×1368 *pixels represented by an image where AD =1712. The subimages "su" are of* 256×256 *pixels. These subimages are cropped from "IM".*

Since the images were going to be cropped, a FOV of 10cm $\times$ 12.5cm was used instead of 3cm $\times$ 3.5cm which other workers have used [92]. This size is justified because statistical features were to be extracted and statistics needs more data.

The optimisation of a FOV involved identifying a feature on a sample. The same feature had to be visually identifiable on the captured image. Additionally, the image had to have minimal gradient illumination and thus a compromise had to be struck between having a saturated image and an image with a poor signal to noise ratio (SNR).

The benefit from cropping is an increase in computation speed as a smaller image has fewer pixels to be manipulated. The cropping of 5 subimages from each sample gives a full representative coverage of the characteristics of the paper. Cropping also boosted the number of original samples which assists in training neural networks. A neural network is a statistical model and hence the need for adequate data.

A handicap in capturing the image is with lighting as the illumination was from AB to CD, thus the upper subimages, $su1$ and $su2$ have saturated pixels whilst $su3$ and $su4$ are likely to have a poor SNR (signal to noise ratio). However, the high pass filter will emphasise the effect. Another solution to this could have been to crop subimages across the centre of the image. Thus only 3 subimages could be cropped from each image instead of 5 subimages.

At the start of the experiment there were 85 large (bigger than an $A3$ size) original samples.

Then 4 samples of $A4$ size were cut from each of the 80 samples, giving a total of 320 samples. The total number of $A4$ size samples cut from the remaining 5 samples was 17 as these were not that big. Consequently, the final database had 337 samples.

The image captured from each sample was cropped into 5 subimages. Instead of having only 337 images, after cropping the total rose to 1685 images albeit somewhat correlated.

The correlation between subimages was overcome through splitting the data in the form of features into 3 separate data files where the first 20 were allocated to training, the second 20 to validation and the third 20 to testing and the process was repeated cyclically until all the data were allocated. The number 20 comes from 4 samples from each original sample multiplied by 5 subimages cropped from each image. This randomisation of the training data ensures that the trained network output does not vary with the order of pattern presentation. However, randomising the vectors is not necessary for the test and validation data sets as these are always presented in batch mode.

There was a good sample-to-sample (from worst to best samples) distribution for the original samples. This guarantees the true nature of the data and better statistics as it offers the best avoidance of correlation between subimages. However, because 4 samples cut from each sample were added to the pool of samples this meant that it was going to be difficult to totally eliminate correlation between the samples. The ideal approach for correlation avoidance would be to collect more than 85 different samples (typically 700). Otherwise, the splitting strategy used above appears to be intuitive and potentially the best randomisation approach for this data.

As previously described at the beginning of this chapter, the paper samples were pre-classified in terms of surface appearance quality. The categories of paper comprised the good, the average and the poor class.

7.1.3 Results from Preprocessing Paper images

The results that follow are from images that were categorised as "poor" and "good" and the aim is to show the importance of the high pass filter for this work. The images in Figure 7.6 were not high pass filtered, consequently, the gradient illumination due to angular illumination from image capture is evident. In Figure 7.7 images have been high pass filtered and there is a stark difference in appearance with the images of Figure 7.6. The high pass filter is therefore useful in this work. Figure 7.3 shows several routes that were followed when preprocessing an image

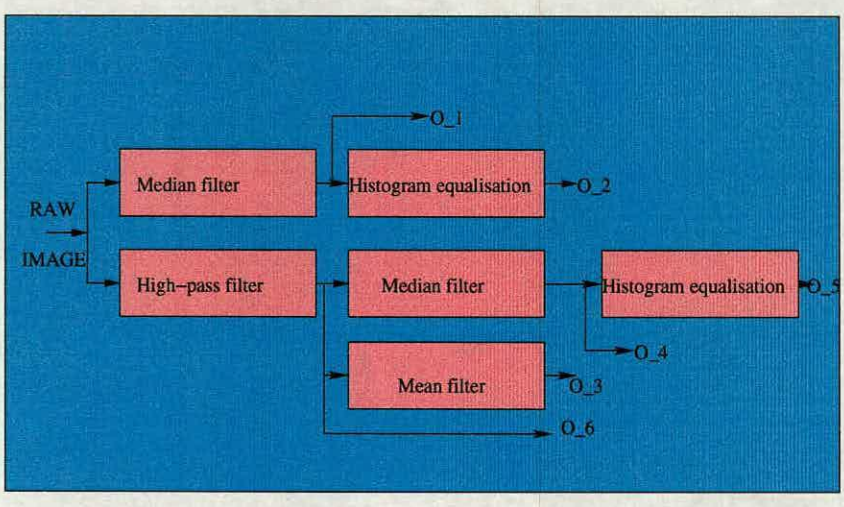
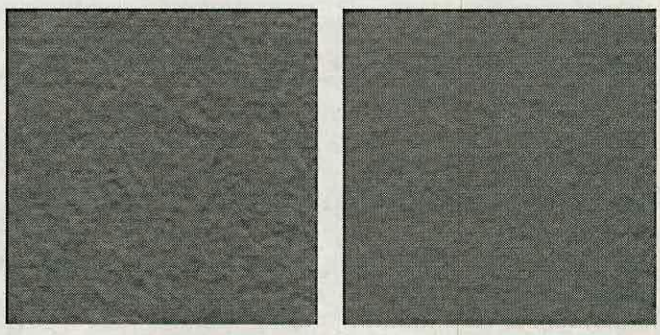


Figure 7.3: *This diagram shows several preprocessing routes.*

and the final choice was O_4. This is because in chapter 2 it was found that the high pass filter can be useful in removing low frequency signals like gradient illumination. Additionally, the median filter was recommended for use in removing high frequency noise. There is no evidence that suggests that there can be impulsive noise, however, since the images are not captured on line, there is a likelihood of dust particles or some small debris settling on the paper and hence creating high frequency noise. It is thus instructive to take precautionary measures since such noise can severely distort results.

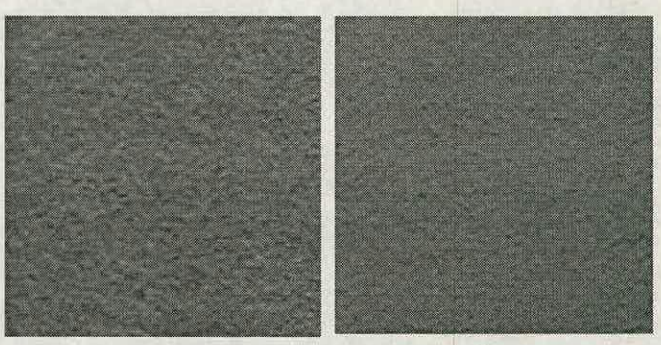
7.1.4 Optimisation of the kernel sizes

The optimisation of the median filter kernel size was implemented through an experiment using separate features from the co-occurrence matrix and the specific perimeter method (from chapters 3 and 5 respectively). Different mask sizes for the median filter were chosen and applied on images classified by the human expert as "poor", "average" and "good". The masks used included the 1×1, 3×3, 5×5 and 7×7 size. The 3×3 median filter's performance approached that of the optimal 5×5 median filter as shown in Figure 7.8. The 3×3 median filter deals with noise less effectively, but detail is better preserved. The 3×3 and 7×7 are shown in chapter 7 in Figure 7.7. However, because the former is closer to noise, it was rejected so is the 1×1 filter. The 7×7 median filtered images in Figure 7.7 were more blurred and were also rejected.



bad image good image

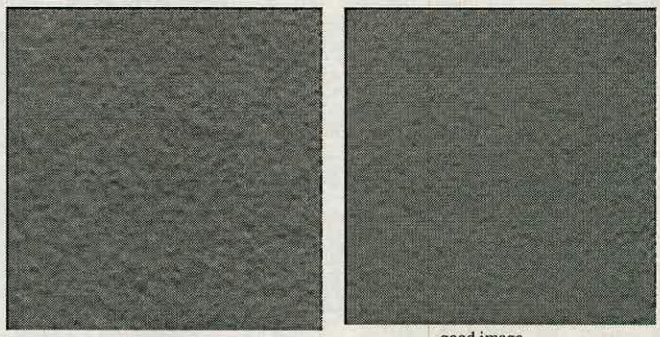
(a) These images are results from filtering images from a "good" paper sample and the "poor" paper sample image. The filter used is a mean filter.



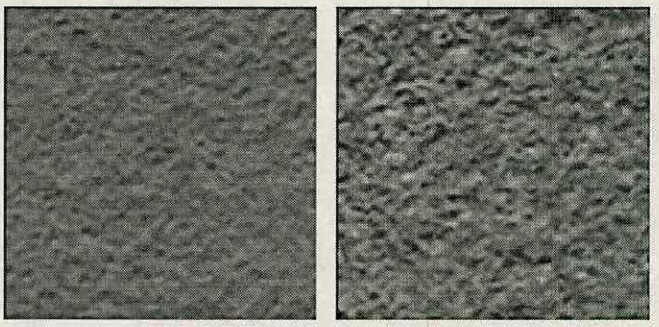
poor image good image

(b) These are high pass filtered images from a "poor" sample and a "good" paper sample. This is illustrated by O_6 in Figure 7.3

Figure 7.4: *Figure 7.4(a) and Figure 7.4(b) are meanfiltred and highpass filtered images respectively.*



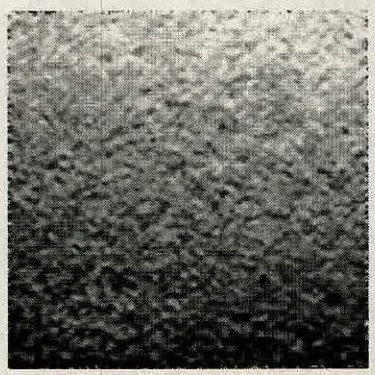
(a) These are median filtered images from a "good" paper sample and a "poor" paper sample. This is illustrated by O_1 in Figure 7.3



(b) These images were high pass filtered, median filtered and then histogram equalised. The images were from "poor" and "good" paper samples. This is illustrated by O_5 in Figure 7.3

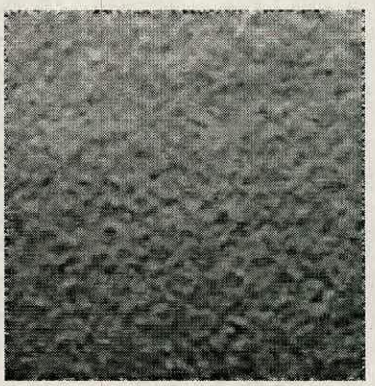
Figure 7.5: *These are filtered poor and good images.*

(a) This is an original image that was categorised as poor.



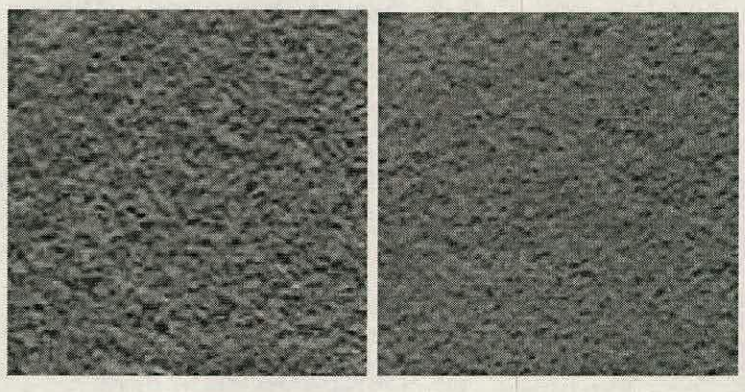
(b) This is an image categorised as poor. It was median filtered and then histogram equalised using 7×7 mask sizes. This is illustrated by O_2 in Figure 7.3

(c) This is an original image that was categorised as good



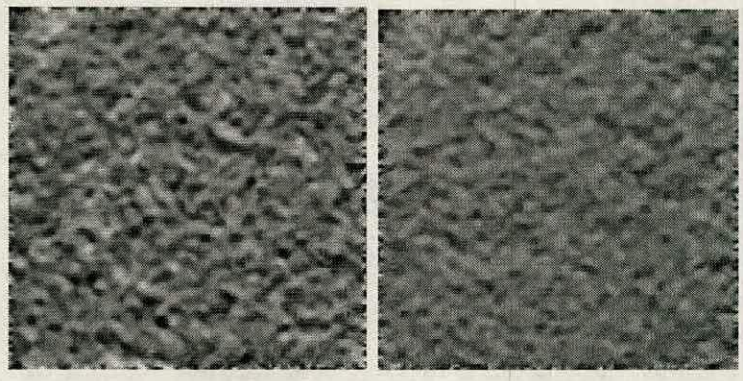
(d) This is a median filtered and then histogram equalised good image (not high pass filtered) using a 7×7 mask size. This is illustrated by O_2 in Figure 7.3

Figure 7.6: *The* Images *have been median filtered and then histogram equalised using* 7×7 *mask sizes.*



(a) This is a high pass filtered, median filtered and then histogram equalised poor image using 3×3 mask sizes

(b) This is a high pass filtered, median filtered and then histogram equalised good image using 3×3 filter sizes



(c) This is a high pass filtered, median filtered and histogram equalised poor image using 7×7. mask sizes

(d) This is a high pass filtered, median filter and then histogram equalised good image using a 7×7 mask size.

Figure 7.7: *These images are a result of a procedure illustrated by O_5 in Figure 7.3*

The co-occurrence matrix features were extracted from the image after filtering with a chosen mask size. A "difference" in value between these feature values from a) the "poor" image and the "average" image, b) the poor image and the good image, c) the average image and the "good" image was computed. The graphical plots shown in Figure 7.8 are for the "difference" in value between the feature values from the "poor" image and the "good" image. The same procedure was repeated using the modified specific perimeter feature. It turned out that the median filter apart from eliminating the high frequency noise enhances the interclass distance.

The results shown in Figure7.8(a), Figure 7.8(b), Figure 7.8(c), Figure 7.8(e) and Figure 7.8(d) are from the co-occurrence matrix features. The results shown in Figure 7.9(a) and Figure 7.9(b) are from the modified specific perimeter features. These graphs show the "difference" in feature value between the "poor" and the "good" image. The results from these graphs suggest an optimal size of 5×5 for the median filter kernel. They also show that the optimal lower bound for the bounding box for the specific perimeter method is 5×5.

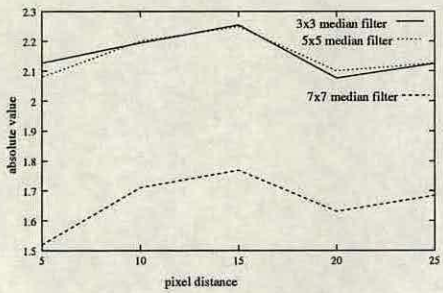
The desired result is a large difference between the two extreme classes, the "poor" and the "good" image and as a consequence, a simple linear classifier can be used. A small difference in feature value was expected from the computation of the "difference" between the image from the "average" class with either of the 2 classes.

The spatial distribution of the grey levels in the window should be symmetrical otherwise the overall mean intensity of an image is altered by the median filter. The median filtered images are shown in Figure 7.5(a).

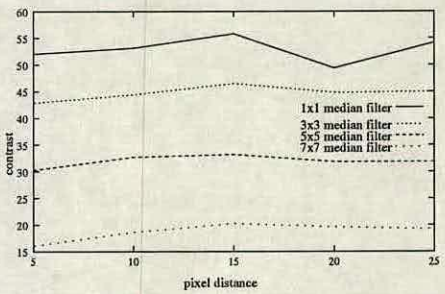
The desired result is a crisp, high contrast image which exposes image features that are critical to the success of the future stages of processing like feature extraction. A suitable median filter size must remove noise that contributes to class overlap and thus give a maximum "difference" in value between different classes of data. In terms of detail preservation, median filtering is superior to histogram equalisation as the latter blurs the image.

7.1.5 Optimisation of the Spatial Grey Level Dependence Matrix parameters

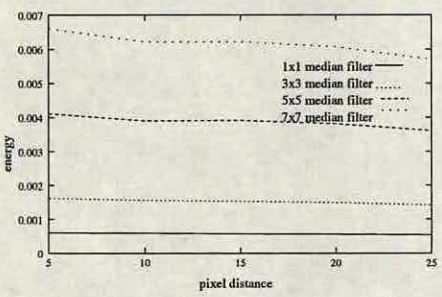
The optimisation involved experimenting with different pixel distances d. The SGLDM features were extracted for each d and used in 10 classification experiments as inputs to a multilayer perceptron (MLP) neural network trained using the backpropagation algorithm with different classifier weight initialisations. The mean for the 10 classification results was taken as the final



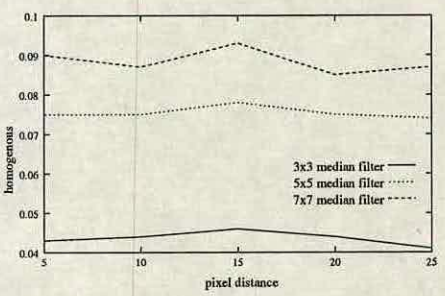
(a) Each graph shown is a result of the "difference" between the absolute value for the "poor" image and the "good" image for a given median filter size.



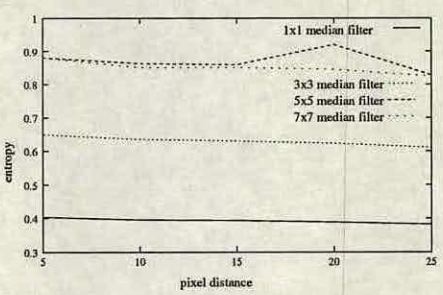
(b) This graph shows the "difference" in value between the contrast value for the "poor" image and that of the "good" image for a given size of the median filter.



(c) This is a graph of the "difference" between the energy value for the "poor" image and that of the "good" image for a given median filter size.

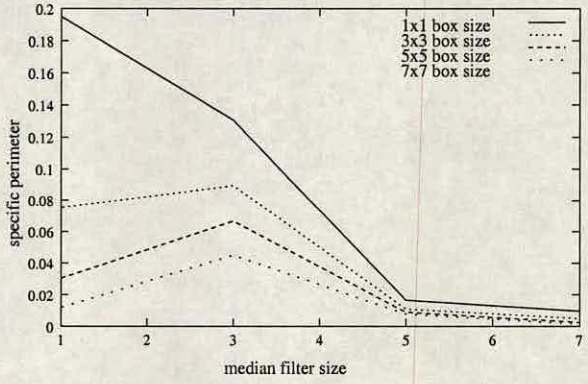


(d) This is a graph of the difference between the homogeneous value for the "poor" image and that of the "good" image at a given size of the median filter.

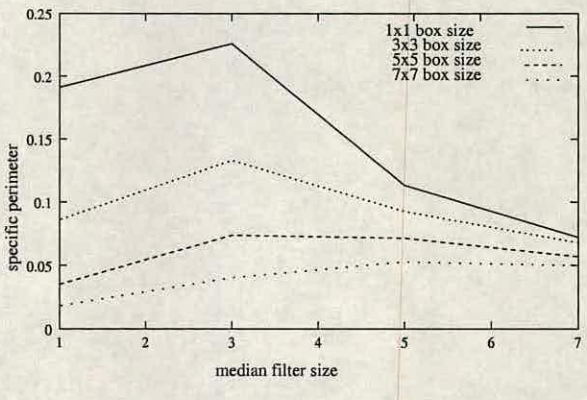


(e) This is a graph of the "difference" between the entropy value for the "poor" and that for the "good" image for different median filter sizes.

Figure 7.8: *The graphs present the results of experiments carried out to optimise the median filter mask size. Each graph shown is a result of the "difference" between the feature value for the "poor" image and the "good" image for a given median filter size.*



(a) good image



(b) poor image

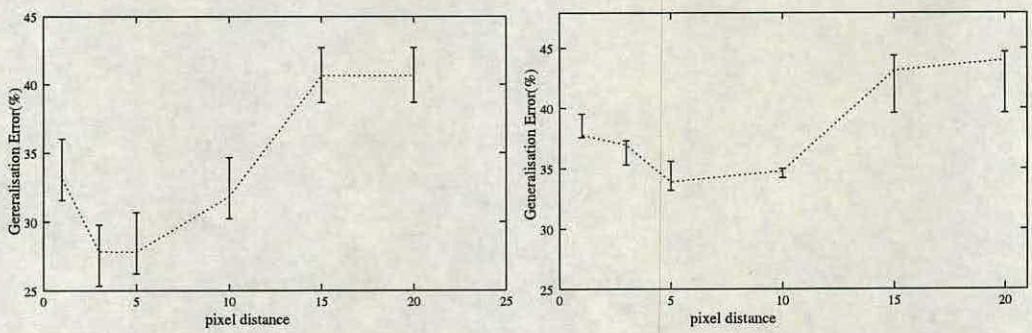
Figure 7.9: *These graphs show the difference between the good and poor image using the Specific perimeter method. Figure 7.9(a)is the plot for SPM for the good image. Figure 7.9(b) is the plot for the poor image. These graphs further show the optimal parameters for the median filter size.*

generalisation error.

The optimal pixel distance for the 2-class and the 3-class problem was the distance corresponding to the lowest generalisation error and in this case it was the 5 pixel distance.

The pixel distances were within the desired range of the features of interest. The SGLDM therefore became sensitive to the feature size of interest. The pixel distances investigated were from 1 to 20 as indicated in Figure 7.10. The 1 and 3 pixel distances were investigated only out of curiosity, otherwise anything outside the 5 to 20 pixel range was rejected as it could have resulted from[53][54] noise and handling respectively.

The graphs Figure 7.10(a) and Figure 7.10(b) are results that assists in obtaining the optimal pixel distance d for the SGLDM matrix. The graph Figure 7.10(b) for the 3-class problem has a tighter error bar at 10 pixel distance (error bar is twice less than that at the 5 pixel distance). However, the 5 pixel distance is close to the minimum feature size and its error bar is tolerable in comparison to other distances. Thus $d = 5$ was chosen as the optimal distance.



(a) This is a graph showing the 2 - class problem

(b) This is a graph for the 3 - class problem

Figure 7.10: *The graphs show the optimal* SGLDM *pixel distance corresponding with the lowest generalisation error at* 5 *pixel distance*

The 15 and 20 pixel distances in Figure 7.10(b) have higher generalisation errors and large error bars hence their results are unstable as indicated by Figure 7.10(b). The error bars for 1 and 3 pixel distances are almost equal but 3 has a smaller generalisation error.

In Figure 7.10(a) all the error bars are almost of the same size and $d = 5$ pixels gives the lowest generalisation error.

The optimisation of the direction θ was performed as follows:
The SGLDM for each chosen pixel distance was computed for each of the 4 directions and then the results were used as inputs to a neural network. The process was repeated for other pixel distances until all the distances were used.

The direction θ whose generalisation error was consistently low was chosen as an optimal direction for which features were to be extracted. It turned out that the 45° and 135° directions consistently produced the lowest generalisation error.

In summary, by varying parameters θ and d, the SGLDM can be calibrated to a range of different textures. The procedure is repeated for all possible pairs of grey level values in the image.

The generalisation error results for the 0°, 90°, 135° directions of the SGLDM. These results show a high performance by the 135° direction and they are similar to that of the 45° direction.

Distance pixels	0°	90°	135°
d1	32.53	26.62	13.4
d3	32.99	26.07	13.8
d5	29.01	25.47	13.6
d7	29.93	26.33	13.57
Combined	29.12	25.00	12.86

Table 7.2: *The SGLDM generalisation error results from varying the direction and the intersample distances. "Combined" are results from intersample distances d2, d3, d5 and d7 combined.*

7.1.6 Optimisation of the Modified Specific Perimeter parameters

The key parameters in the specific perimeter method (SPM) are the field of view (FOV), the thresholding method and the aspect ratio (the ratio of the width to the height of a blob). The SPM decreases exponentially with the increase in the resolution and the FOV [92]. In this thesis we seek to optimise the FOV. The FOV that is larger than the scale of variation [92] (fibre length) can lead to a higher local threshold (median) value which can exclude essential blobs and hence a loss of fine detail could result in a low SPM value. This is a consequence of pixels being further apart physically and in their grey level values within the same neighbourhood. Jordan [92] has shown that the SPM is relatively insensitive to the FOV that is larger than the fibre length.

The median as a threshold and not the mean is useful for application on paper images because it is robust to high frequency noise whereas the mean assumes a very high or very small value and thus pixel values which are representative of the true surface appearance of paper are excluded. The noise could have resulted from foreign particles and not the process that produces paper.

The optimisation involved experimenting with different bounding box sizes when extracting specific perimeter features. A bounding box is a window that borders a blob and it thus filters out blobs that fall outside a given range. The SPM feature computed from images filtered by each bounding box size were used in 10 classification experiments with different classifier weight initialisations. The mean results were used as the final result. The SPM feature was used as an input to a multilayer perceptron (MLP) neural network trained using the standard backpropagation algorithm. The results shown in Figure 7.11 and Figure 7.12 are plots of the generalisation error against the box size. The optimal bounding box size corresponds to the lowest generalisation error.

The modified SPM was implemented such that blobs that are outside the desired range and "noise" were filtered out. Thus for the surface appearance characterisation, the optimal aspect ratio, the minimum and maximum box size were obtained through experimentation. The SPM therefore became sensitive to the feature size of interest. Whereas during experimentation as indicated in Figure 7.11, the bounding box sizes that were used were from 1 up to 20, in the final analysis the bounding boxes outside the 5 to 30 pixel range were rejected as they could include noise or creases due to handling respectively.

7.2 Classification

This section presents classification experiments carried out on manufactured paper. The experiments were split into two: one was for classifying the two extreme classes, the "good" and the "poor". This experiment shall be referred to as the "2-class problem" . The other experiment was for classifying the "good", the "average" and the "poor" class. This problem will be referred to as the "3-class problem". The third part of the experiment involved reducing the most dominant class so that the number of samples in it almost equal the number of samples in the other 2 classes.

The "2-class problem" was used for finding potential features for the "3-class problem" which was the desired network. This strategy was motivated, from a visual inspection of samples

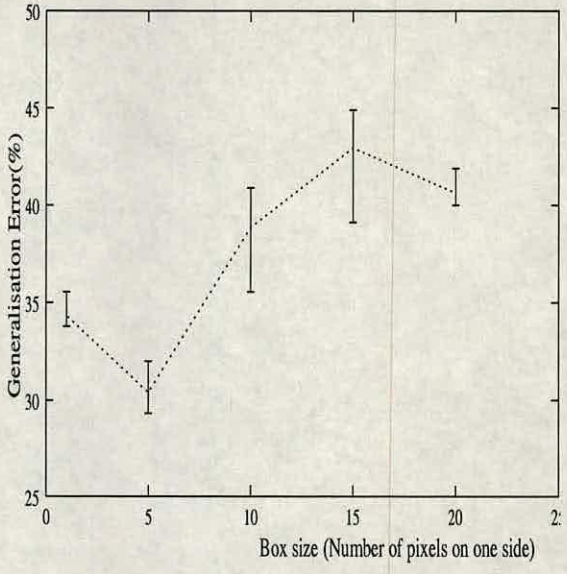


Figure 7.11: *This graph shows the results from an experiment to optimise the box size for the specific perimeter method. The classification was done using only 2 classes, the "good" and the "poor" classes.*

perspective, by an overlap of data samples between the "average" class with either the "good" or "poor" classes. In addition, the "2-class problem" had the computation speed advantage.

exp	exp	train	valid	test	total samples
2-class	**exp1**	230	225	225	680
3-class	**exp2**	565	560	560	1685
average-class reduced	**exp3**	410	408	408	1226

Table 7.3: *This is training, validation and test data and the total number of samples used in each experiment*

The data set for each of the three experiments reported in this chapter was split into three: for training, validation and testing as shown in Table 7.3.

7.2.1 Experiments for the 2 class problem

Two categories of paper were used: the "good" and "poor". Classification was carried out using both a *linear classifier* and the most commonly used (*non-linear classifier*) one hidden layer multi-layered perceptron (MLP) neural network. The linear classifier was used in order to confirm our earlier conclusion that the data is non-linear. Both classifiers were trained using

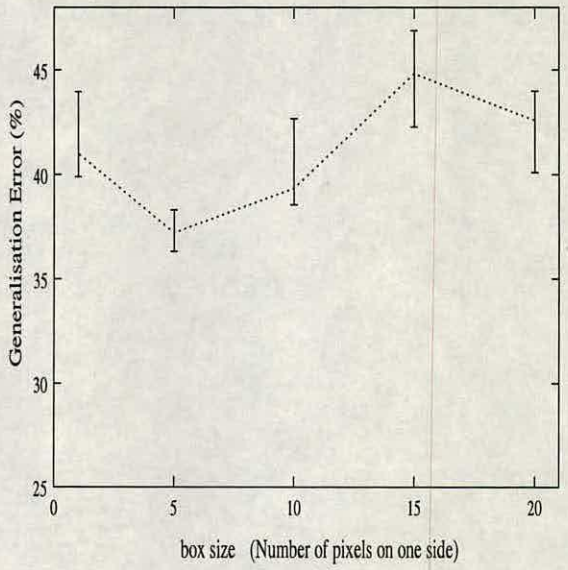


Figure 7.12: *This graph shows the results from an experiment to optimise the box size for the specific perimeter method. The classification was done using all the 3 classes, the "good", the "average" and the "poor".*

the standard gradient descent method and 10 training runs were performed for each chosen network. Training was stopped when a minimum was reached in the validation error. The test set which is part of the collected data was used to evaluate the generalisation performance of a trained neural network. The different stages of training were shown in Figure 6.6 in the previous chapter.

The reason for 10 training runs was to ensure that a statistically significant result was obtained and hence the final architecture and a set of weights obtained will be nearly optimal. The optimal networks were those that recorded the lowest minimum generalisation error on the validation set. In varying the number of hidden units an upper limit was put at 30 hidden units otherwise longer training times and also overfitting of the training data occurs.

7.2.1.1 Results from Lacunarity and Fractal dimension

The results from the FD and lacunarity on the 2-class problem were satisfactory as shown in Figure 7.13. The generalisation error of 41% renders these techniques useless in building a paper classification system.

The lacunarity and the FD features were rejected because the roughness information provided

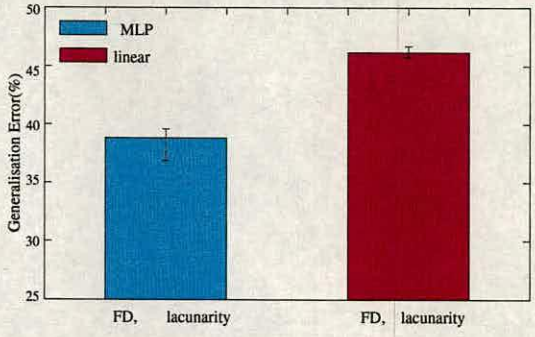


Figure 7.13: *This Graph shows the results from the fractal dimension and lacunarity taken from the 2-class problem.*

by the FD was found to be almost 84% (result indicated in Table 7.4) correlated to the grainness information of the modified SPM. This result was obtained by passing the SPM and the FD through a correlation algorithm.

	spm	**FD**
spm	1	0.838
FD	0.838	1

Table 7.4: *This is a table for the correlation results between the SPM and the FD*

In terms of further study, the FD in combination with the Discrete Fourier Transform should be explored with the former giving a coarseness measure and the latter giving the directionality information. Direction in a textured surface of an image is explained as texture that progressively increases in coarseness from fine to coarse texture or vice-versa.

7.3 Optimisation of the Gabor parameters

This section gives detail on the optimisation of the Gabor filter parameters. These parameters were covered in chapter 4 and the key parameters in this context comprise the window size (standard deviation σ) and the direction θ. The frequency parameter is irrelevant in this context as the images were captured in the machine direction and texture in this direction has a random pattern. The recipe for the optimisation of these parameters is follows:

- choose a direction θ

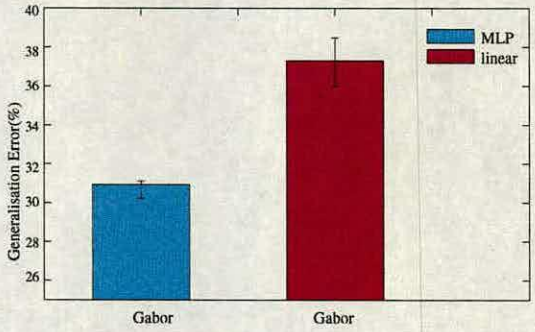


Figure 7.14: *This Graph shows the results from the Gabor filters at 45 degrees for the 2-class problem.*

- choose the size of windows (σ)
- compute Gabor energy for each window until all windows have been used
- choose another direction θ
- repeat the process until all θ have been used

In each case the feature corresponding to each window size and θ was used as an input to the classifier and the generalisation error noted. The window and the direction that gave the lowest generalisation error were chosen as the optimum parameters. As an additional constraint, the window size had to be at least larger than 3×3 otherwise the results could be confused with noise and edge effects.

The optimal window size and direction were 5×5 and 45° respectively. Table 7.5 is the summary of the results from the optimisation of the Gabor parameters. Here only the summary of the best results from optimal parameters corresponding to the optimal 5×5 window is given.

Gaussian window	0	45	90	135
MLP	47.33	38.5	44.2	38.96
Linear	53.45	41.78	51.33	43.44

Table 7.5: *This is a summary of results using features computed from Gabor images. The optimal* 5×5 *window size was used.*

The FOS features extracted from a Gabor filtered image comprise the mean, energy, entropy and variance. These features were used as an input to an MLP and a linear classifier. Just like other classification experiments reported in this work, 10 classification runs for each architecture

were performed and the result with the lowest generalisation error was chosen. A combination of features obtained from all the windows ($1cm$, $3cm$, $5cm$ and $7cm$) and directions θ (0°, 45°, 90°, 135°) were also used as an input to the classifier. The 31.5% generalisation error as indicated by Figure 7.14 for the Gabor filters is satisfactory.

7.3.1 Optimisation of the DCT parameters

This section describes optimisation of the DCT parameters. The implementation involved computing DCT for window sizes of 2×2, 4×4 and 8×8. The features from the resulting DCT images were used as inputs to the MLP classifier and the linear classifier. Then 10 classification runs for each architecture were performed and the result with the lowest generalisation error was chosen. The window size that performed better is the 8×8. We could have gone for the 16×16 but then computation becomes very expensive and the increase in performance is insignificant. It is well known now that window sizes above 8×8 do not give a significant improvement to the result. The DCT features were then computed once the optimal size of the mask had been found. The features computed included the energy feature.

The feature that was extracted from the DCT is the energy feature. The 33.7% generalisation error shown in Figure 7.15 for the DCT is satisfactory. The DCT fares well in images with edges and lines, yet in manufactured paper these features are less pronounced, hence its performance.

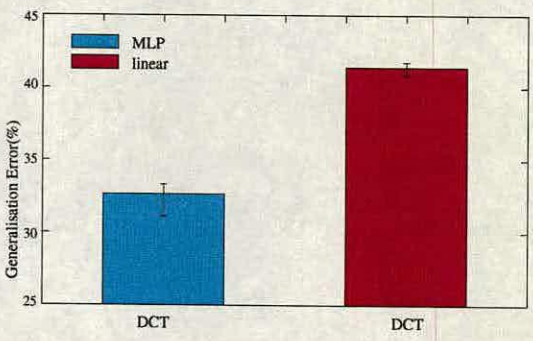


Figure 7.15: *This Graph shows the results from the DCT for the 2-class problem.*

The linear classifier using either of the features (DCT or Gabor) performance was low because the data is non-linear. The combination of these features with those from the spatial techniques for the 2-class problem gave a generalisation error of 37% which is very poor compared to the generalisation error of 13% that was obtained from the spatial techniques alone..

7.3.2 Classification using Spatial Techniques

Classification was initially performed using the SGLDM's energy, entropy, contrast, homogeneity, absolute value; the GLRLM 's run length non-uniformity, short run emphasis, long run emphasis, grey level run length non-uniformity, run percentage; the SPM'sas indicated by Figure 7.14. specific perimeter and blobs and the FOS's kurtosis, skewness, mean, standard deviation and the variance. The classification results are shown in Figure 7.16.

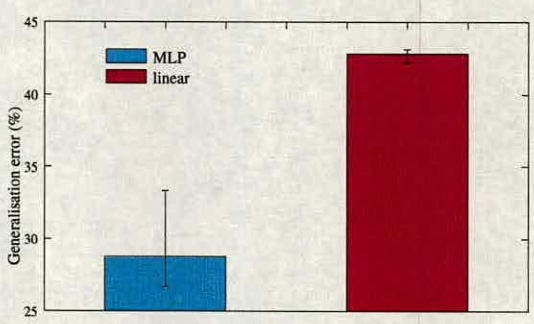


Figure 7.16: *This Graph shows the results from the* 17 *features taken from the SGLDM, FOS, GLRLM the and SPM*

These features were chosen using the knowledge of the methods (considering the texture they assess) and the knowledge of the problem. The genetic algorithm (GA) although it is equally good, the problem is that if an additional feature were to be added in the future, getting an optimal mutation is not easy. Additionally, the GA takes a long time to reach an optimal solution. This is a disadvantage since if in the future a feature is to be added, then it will have to re-search the feature space for an optimal solution. In contrast, in the manual method only needs re-training the classifier for any additional feature added.

As an initial step for the manual approach, an off-the-shelf correlation algorithm was used to compute correlation between the features. Features from each technique were passed through a correlation algorithm at a time. The expected output is 1 along the diagonal as a feature is correlated to itself. If the correlation between two features is 0.7 upwards, these features were viewed as being correlated. One of the correlated pair was removed from the set

This procedure was performed first on features from individual techniques. The next stage involved computing correlation for the whole set of available features from multiple techniques.

The 17 features were passed through a correlation algorithm and the results noted. Only non-

correlated features were selected. Added to these features are features selected from intuition based on the knowledge about what they sense from the data irrespective of whether they were correlated to the features already included. This knowledge is coarseness, directionality, linearity, regularity and roughness of the paper surface. This combination of features shall be called "Combined 12" and the features that were used are:

- the short run emphasis,
- the long run emphasis,
- grey level non-uniformity
- run length non-uniformity for the GLRLM;
- entropy,
- contrast,
- absolute value
- energy, for the SGLDM;
- specific perimeter
- the number of blobs in the SPM; and
- first order statistics,
- standard deviation and kurtosis.

Classifiers were trained and tested using these features in combination and the results are labelled "combined 12". The combination of features within each technique were also used to characterise paper and the results are shown in Figure 7.17.

The 3rd experiment involved reducing the number of features by assessing the inter-feature correlation. Those that remain are:

- for the GLRLM
- short run emphasis,

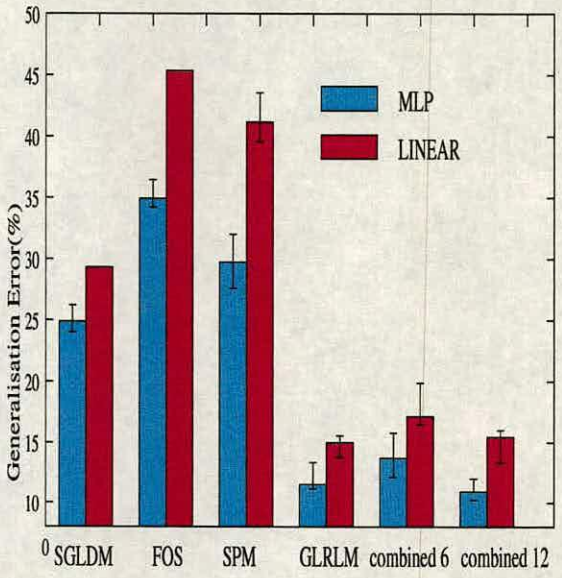


Figure 7.17: *This Graph shows classification results for the 2-class problem for surface appearance. For each pair of plots, the taller and short plots are results from linear and non-linear MLP classifiers respectively. The error bars indicate the maximum and minimum classification results.*

- long run emphasis and
- run length non-uniformity
- for the SGLDM:-
- entropy and
- energy
- and the SPECIFIC PERIMETER.

Reducing the number of features can result in a slight increase in the generalisation error and the benefit is a gain in the classification speed resulting from few inputs to the classifier. A reduced feature set can also lead to a decrease in the generalisation error and in this case removed features are viewed as noise.

Classification was performed as in the previous experiments using these six parameters and results obtained are labelled "combined 6" in Figure 7.17.

The 4th experiment was carried out on "Combined 12" features using the principal component analysis to select an optimal set of features. The principal component analysis was covered in chapter 6. The 12 features were used as an input to the principal component algorithm. The process is illustrated by Figure 6.3. Since the focus was reducing the input to the classifier, the principal components that included 3, 4, 5, 6, 7 and 8 represented as PCA3, PCA4, PCA5, PCA6, PCA7 and PCA8 respectively were selected. Using any PCAs beyond PCA8 would be as good as having used all the 12 principal components hence defeating the purpose of feature reduction. Each of these PCAs were used as inputs to both the linear classifier and the MLP classifier and classification was performed in a similar fashion as in the previous sections. The results are shown in Figure 7.18.

PCA7 and PCA8 have a higher dimensionality, have far less tight error bars than that for PCA6, but they have lower generalisation errors than PCA6.

In the low dimension group that comprises PCA3, PCA4 and PCA5, PCA3 is the best among these despite having the worst generalisation error, it has a tight error bar and thus its result is stable. PCA5 is the worst in terms of stability as it has the largest error bar among this group.

For PCA6, PCA7 and PCA8, PCA6 is the best set despite its slightly lower generalisation error compared to that of PCA7 and PCA8. Its error bar is tight and that compensates for its lower generalisation error. Furthermore, its other advantage is that it has a lower dimension in feature space than PCA7 and PCA8 and this increases the classifier speed. Overall, PCA6 gives the best compromise of stability and generalisation performance.

The performance by the PCA was marginal because it is a linear technique whereas the data is non-linear.

In terms of the generalisation error, the modified SPM ranked third from GLRLM and SGLDM as shown in Table 7.6. However, the modified SPM uses a smaller number of hidden units hence classification is faster than when the other two techniques are used. In terms of cost, the SPM incurred the highest cost whereas the GLRLM incurred the lowest cost. Cost in this context, is the risk associated with misclassifying a sample of paper. There is a section wholly devoted to cost later in this chapter.

Features used from different techniques in combination, the *"Combined 12"* and the *"Combined 6"* used more weights than any of the individual techniques. In terms of cost the "Com-

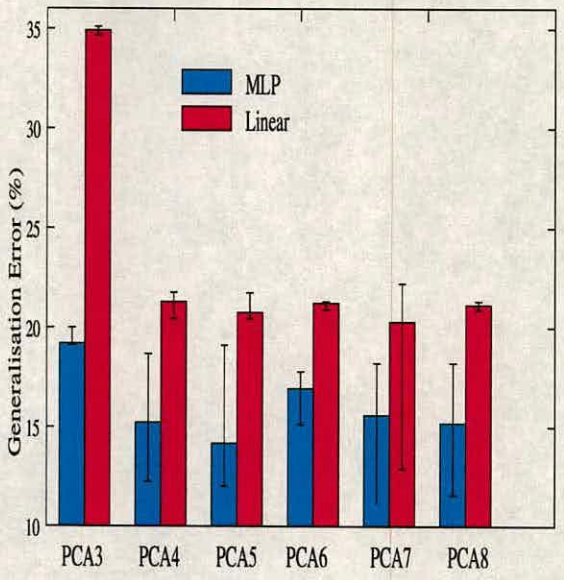


Figure 7.18: *The Graph shows classification results from the PCA for the 2-class problem for surface appearance. The error bars indicate the maximum and minimum results. Among the pair of bars, the short represents generalisation by the MLP whereas the longer represents that by the linear classifier.*

bined 6" incurs almost twice the cost incurred by "Combined 12" as shown in Table 7.6. Combined 12's generalisation performance is higher than any of the strategies used. Whilst "Combined 12" and "Combined 6" have the same number of hidden units, classification using "Combined 6" took a third of the time taken by "Combined 12" as the number of inputs to the classifier is half that of "Combined 12".

7.3.3 Summary for the 2-class problem

"Combined 12" from intuition plus correlation achieved a low cost and high classification performance compared to "Combined 6" obtained from non-correlated features. These results are shown in Figure 7.17 and Table 7.6. The latter has fewer features hence the classifier trained on it is much faster.

The cost for "Combined 6" was expected to be higher as the reduction of features is accompanied by an increase in the uncertainty in the decision space. Its slightly lower generalisation error was expected and as in many cases, a subset of features should not perform better than a full set that includes it.

Techniques	generalisation and risk		
	class	cost	architecture
Combined 12	87.41	64.24	12 25 2
GLRLM	86.70	122.08	4 20 2
SGLDM	76.67	126.42	4 8 2
GLDM	73.33	150.01	4 12 2
SPM	69.67	247	2 8 2
FOS	64.88	193.98	2 2 2
Combined 6	85.60	104.88	6 25 2

Table 7.6: *This is a summary of results that include cost for the 2 class problem. The cost is defined in section 7.2.3.2. The architecture comprises, the number of inputs, the number of hidden units and the number of outputs respectively.*

		classified as	
		Good	poor
Actual	Good	40.000	4.444
	poor	8.000	47.556

Table 7.7: *The confusion matrix for the combined best features (Combined 6) from the data shown in Table 7.3*

The results show that features obtained through the knowledge of the information (the power of intuition over using only non-correlated features) they capture from texture irrespective of them being correlated, can improve classification performance. This is already a commonly held view that some of the correlated features might carry slightly different but important discriminatory information thus great care must be taken before they can be rejected.

The results have been interpreted in terms of cost and complexity (number of hidden units). The desired result must achieve a high generalisation performance, a low cost and use few hidden units. It turns out that the GLRLM out-performed other individual techniques in terms of generalisation error and cost. The FOS had the worst performance than the spatial techniques used.

The results due to features selected using the PCA approach in Figure 7.18 are lower than those of Figure 7.17 obtained using the knowledge of the features plus correlation ("Combined 12"). This was expected as the PCA is a linear technique whereas the data is nonlinear.

In comparison, the manual and the PCA feature selection approaches, the former outclassed the latter by more than 5% (comparing "combined 12" with PCA). In terms of speed it is faster to

implement than the manual approach. Where accuracy is not critical, then the PCA might be suitable. In this context, the manual approach is favoured.

The time taken to compute the result during training ranged from 120 minutes to 240 minutes depending on the architecture of the classifier, epochs and the size of the PCA (number of outputs). The performance in terms of run-time was 30 seconds for a trained network. Since classification is not performed in real-time, this time is acceptable times in the plant.

The features from different techniques in combination obtained a better coverage of the discriminatory information than when only features from one technique are used. This optimal discriminative set of features might be useful in building a classifier for the *" 3-class problem"* covered in a later section.

7.3.4 Classification for the 3 - class problem

This section presents confusion matrices in conjunction with the loss matrix [134]. These approaches are useful in evaluating classification results. The results from the 3-class problem are also presented in this section.

7.3.4.1 Confusion matrices

The confusion matrix shown in Figure 7.7 gives both the information on how the network performs over the entire test set and the point where errors occur. In the latter case relevant data can be collected to rectify the error. The confusion matrix's rows "i" and columns "j" represent the classification by the expert (the author of this thesis) who labelled the data and classifier outputs respectively and they are represented by "actual" and "classified as" respectively. Each entry in this matrix is the number of paper samples of the ith class (grade) for which the classifier outputs as the jth class.

The sum of the entries in the leading diagonal is the generalisation result of the classifier hence the sum will be 100% for a perfect classification (with zero entries for the rest of the matrix).

7.3.4.2 Cost

The cost is the risk incurred in misclassifying data. The cost of misclassification can be assessed by defining a "**loss matrix**" [134] also shown in Table 7.8. This matrix converts a probability

result (classifier output) into a decision by computing the cost of each classifier output. The classifiers in this thesis so far have been trained to minimise classification error. However, it is better to train them to minimise the cost to the papermaker.

It is more costly in papermaking to misclassify a "poor" sample as "good", than to classify a "good" sample as "poor", as in the former case a "poor" sample may be delivered to the customer, while in the latter a "good" sample may be reprocessed as broke which is potentially less costly than Customer dissatisfaction. When a "good" sample is classified as "average", it is to the advantage of the customer and it is less risky as it is the manufacturer who incurs a loss on materials and labour.

		classified as	
		Good	Poor
Actual	Good	0	2
	Poor	12	0

Table 7.8: *This is a loss matrix used for computing the cost.*

Each input-output pair for the neural network classifier is called a pattern. The entries (L_kj) of the loss matrix in Table 7.8 were chosen by hand (chosen by the author of this thesis) and they represent the penalty associated with a pattern that is in the wrong class for paper. In this case a very large number 12 was put where the poor sample is classified as a good sample whereas a small number 2 was put where a good sample is classified as poor. The latter is a lesser risk than the former as in the former the customer is supplied with a poor instead of a good sample.

The recipe for computing cost is as follows:

- compute the confusion matrix of the classification result
- get a suitable loss matrix of the same dimension as the confusion matrix
- compute the cost by multiplying corresponding cells of these matrices and then perform a linear combination of the results.
- Is the result greater than the chosen threshold value? If yes, reject classification otherwise accept it.

We know that 100% correct classification implies non-zero values only on the leading diagonal of the confusion matrix and hence the cost will be zero. The author of this thesis set the

threshold for the cost. The setting of these values is subjective. If in the cell of the confusion matrix that corresponds to the one in the loss matrix that has to do with the customer's entry is a large value, then the result is potentially be a candidate for rejection.

Thus if the cost is higher than a set threshold value, the paper should not be classified. For all the patterns in class C_k, the expected loss is given by [134]:

$$R_k = \sum_{j}^{c} L_{kj} \int_{\mathcal{R}_|} p(x|C_k)dx \tag{7.1}$$

where C is the total number of classes ("good", "average" and the "poor" paper sample classes) and x is a pattern. The overall expected loss (Risk) is:

$$R = \sum_{k=1}^{c} R_k P(C_k) = \sum_{j=1}^{c} \int_{\mathcal{R}_|} \{\sum_{j=1}^{c} L_{kj} p(x|C_k)\} \tag{7.2}$$

This risk is minimised if the regions $\mathcal{R}_|$ are chosen such that each sample x $\in \mathcal{R}_|$. In terms of texture classification, $\mathcal{R}_|$ are the three categories of paper, the "poor", the "average" and the "good" class. The decision rule for minimising the probability of misclassification generalises to:

$$\sum_{k=1}^{c} L_{kj} p(x|C_k) P(C_k) < \sum_{k=1}^{c} L_{ki} p(x|C_k) P(C_k) \tag{7.3}$$

for all $i \neq j$.

The cost is the sum of the partial product of each entry L_kj in the loss matrix with the corresponding entry in the confusion matrix. The desired solution gives a low cost. A simple loss matrix incurs a loss of 1 and 0 if the pattern is placed in a wrong class and in a correct class respectively. However, in terms of paper classification, a slightly more complex loss matrix was designed in Table 7.8 with a customer in mind. Any entry L_kj that has to do with the customer (an entry that corresponds to customer being supplied with a poor quality product) was weighted heavily (a higher penalty for misclassification).

The mse when used on the same distributions which are well separated on the x-axis and heavily overlapped on the y-axis, results in a small and large mse respectively. The mse and cost are linked because if the mse in the classifier results in a higher generalisation error, then the

confusion matrix will have higher entries in the off leading diagonal cells and hence a larger cost will be incurred (product of confusion matrix entries with that of the loss matrix gives cost).

The mse (mean squared error) when used as an error function during training optimises the fit to a dominant sample class and the less dominant classes are viewed as outliers, and classified under the dominant class. This is its major weakness.

The confusion matrix can identify this problem. The use of equal prior probabilities (the number of samples of each paper grade divided by the total number of paper samples) of each grade can solve this problem. In this context equal means reducing the number of samples in the dominant "average" class so that its samples are almost equal in number to those in the good and the poor classes. Thus, this is not an attempt to make the poor samples look like good, but to make decision spaces representative of the classes in the data, thus data which has not been seen before but which closely resembles one of the classes used in training the neural network can be classified correctly.

Another error function, the sum-of-squares error (**sse**), cannot distinguish the distributions having the same variance and mean. Additionally, if prior probabilities differ between training and test sets, the sse is modified by introducing a weighting factor. However, the minimisation of the sse at the network outputs maximises non-linear feature extraction criterion at the hidden units [134]. Additionally, the sse does not require the target data to be Gaussian distribution.

An approach called the **reject option** [134] is a classifier that rejects only the "poor" sample and accepts the "good" sample (when used on paper). The classifier output is the posterior probability $P(C_x|x)$ of the class membership. Its implementation involves setting a threshold (a value of the posterior) θ so that classification is only performed when max $P(C_x|x) > \theta$ otherwise the result is rejected. As θ increases, the number of rejected outputs ($P(C_x|x)$) increases and the percentage error decreases. Doubtful paper images, in this case the images at the boundary of the "average" class with either of the two classes (the good and the poor classes) can be examined and classified by a human expert. This class comes between the good and the poor class samples. Thus at the boundaries of this class with either the other 2 classes, the boundaries are fuzzy.

The number of misclassifications is weak as a measure of performance because it is a binary function and especially when optimisation techniques based on gradient descent are used.

When confidence intervals are used, a tighter error bar represents a statistically significant classification result. A large number of samples being used assists in obtaining a higher confidence in the result. In feature selection, the features selected minimise the overall classification error and not cost. The cost can be minimised by further training the neural network, for example, performing 10 classification runs for each chosen architecture. As for the purpose of further study, instead of using the mse, a training strategy with a function that asks real cost must be used.

7.3.4.3 The 3-class problem

The experiments discussed in this section were based on the "good", the "average" and the "poor" sample classes using the full data set labelled exp2 in Table 7.3. The same procedure used in carrying out experiments for the "2-class problem" was used on the three classes. It turned out that the features used in the "2 - class problem" produced good results for the "3-class problem".

The next experiment involved reducing the number of samples in the dominant "average" class to almost the level of the other classes and its results are shown in Figure 7.19. There had to be a compromise between reducing the number of the dominant class and also ensuring that it does not become too low as a neural classifier is based on statistics and thus it needs more data for mapping decision spaces accurately. As in the previous experiments, the number of hidden units were varied and for each set of hidden units 10 classification runs (with different random initial weight values) were performed in order to get a statistically stable classification result. The variables were the numbers of hidden units, the number of inputs and random initialisations for weights and we had control of only the first two variables. The optimal networks selected were those that attained the lowest minimum classification error on the validation set. The

Techniques	av generalisation		
	class	cost	architecture
Combined 12	73.57	161.37	12 20 3
GLRLM	71.97	162.75	4 25 3
SGLDM	66.78	160.03	4 20 3
SPM	63.56	174	2 8 3
FOS	62.11	166.01	2 8 3
Combined 6	72.36	161.59	6 30 3

Table 7.9: *This is a summary of results obtained from the 3 class problem.*

results in Table 7.6 show that this problem is difficult.

		classified as		
		Good	Average	Poor
Actual	Good	0	1	2
	Average	6	0	2
	Poor	12	8	0

Table 7.10: *The loss matrix used for computing the cost/risk*

		classified as		
		Good	Average	Poor
Actual	Good	14.345	5.119	0.000
	Average	4.404	59.642	0.416
	Poor	0.000	15.654	0.238

Table 7.11: *This is a confusion matrix for the SGLDM*

		classified as		
		Good	Average	Poor
Actual	Good	14.178	5.329	0.142
	Average	4.214	58.107	2.107
	Poor	0.000	14.821	1.071

Table 7.12: *This is a confusion matrix the FOS*

The reason for reducing the "average" class was because of the low entries along the leading diagonal of the confusion matrix in Table 7.15. However, in real life equal classes rarely happen as there is always samples that are misclassified. The reason being that natural data takes a Gaussian distribution and thus towards the tail ends there is bound to be an overlap of classes.

Balanced in this context means having the training set with almost equal sized classes. Depending on the error function used during the training of a classifier, the class with too few data can get classified under the dominant "average" class. Reducing the larger class turned out to be a usable solution as indicated in Table 7.16 and Table 7.18. The conclusion drawn from this is that a balanced data set can improve classification and reduce the cost incurred during classification. However, this problem is difficult (as shown in Figure 7.19) as even with the middle class removed 13% of the patterns were still misclassified at best.

The magnitude of the classification errors was assessed from confusion matrices. These matrices incorporate an interpretation with respect to a classification. A useful solution is a confusion matrix with values only on the left diagonal.

		classified as		
		Good	Average	Poor
Actual	Good	14.404	5.148	0.0895
	Average	3.898	60.178	0.387
	Poor	0.000	15.625	0.267

Table 7.13: *This is a confusion matrix for the SPM*

		classified as		
		Good	Average	Poor
Actual	Good	14.419	6.964	0.044
	Average	3.883	56.428	1.473
	Poor	0.000	15.714	1.0715

Table 7.14: *This is a confusion matrix for the GLRLM*

		classified as		
		Good	Average	Poor
Actual	Good	14.517	6.071	0.179
	Average	3.570	57.910	0.303
	Poor	0.000	16.642	0.143

Table 7.15: *This is a confusion matrix for "Combined 6".*

		classified as		
		Good	Average	Poor
Actual	Good	24.019	4.144	4.631
	Average	3.881	33.235	8.231
	Poor	1.462	8.889	18.367

Table 7.16: *This is a confusion matrix for Combined 6 with the "average" class data reduced.*

		classified as		
		Good	Average	Poor
Actual	Good	14.583	5.059	0.000
	Average	3.869	60.119	0.476
	Poor	0.000	15.774	0.119

Table 7.17: *This is a confusion matrix for the PCA.*

		classified as		
		Good	Average	Poor
Actual	Good	32.683	4.634	9.878
	Average	2.561	11.707	0.366
	Poor	13.415	5.610	21.707

Table 7.18: *This is a confusion matrix for the "Combined 12" with the "average class" data reduced.*

"Combined 12" from intuition plus correlation achieved a low cost and high classification performance compared to "Combined 6" obtained from non-correlated features. The results shown in Figure 7.19 and Table 7.9. "Combined 6" has fewer features hence the classifier trained on it is much faster. However, fewer features can be accompanied by an increase in the uncertainty in the decision space and also a subset of features should not perform better than a full set that includes it.

The results show the power of intuition over using only non-correlated features. Thus not all correlated features must be rejected as they might have slightly different discriminatory information.

The GLRLM out-performed other individual techniques in terms of the generalisation performance. However, it incurred a slightly higher cost and used a larger number of hidden units than the SGLDM. The FOS, again had the worst classification performance.

Features from different techniques in combination ("Combined 12" and "Combined 6") obtained a better coverage of the discriminatory information than when only features from one technique are used.

Thus a balance between the requirement of good performance and computational complexity has been partially achieved. This will become clear in chapter 8 where an extensive summary is given.

An automatic system for the characterisation of surface appearance quality using texture analysis and neural network classification has been successfully built.

7.4 Chapter Summary

The surface appearance of paper has been classified using a nonlinear classifier, the MLP. An approach using texture analysis to optimise the relative physical properties of paper has been developed. Multiple features from different texture analysis techniques in combination improved the classification of paper beyond single-feature analysis. Features selected using the principal components analysis gave an satisfactory performance among the strategies that were employed.

The combination of decorrelated features and features from intuition (hand-picked) gave a good

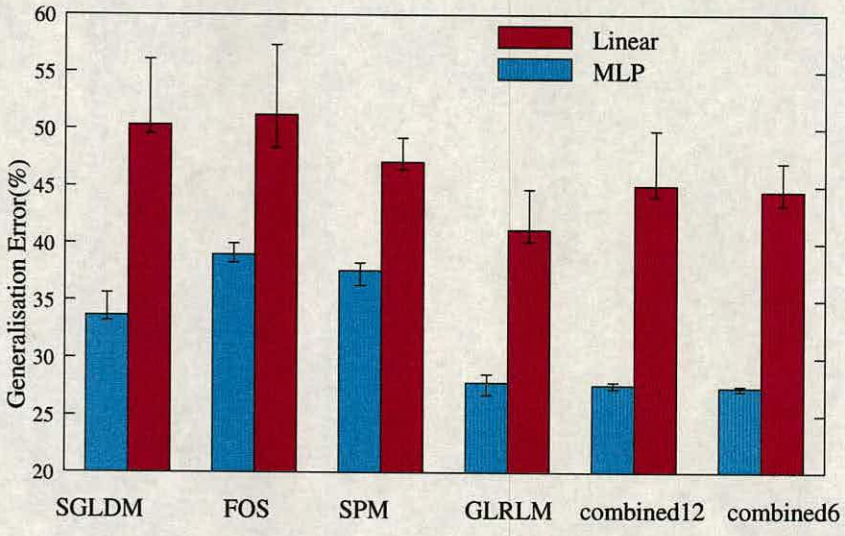


Figure 7.19: *This Graph shows classification results for the 3-class problem for features from individual techniques and for features from different techniques used in combination. The error bars indicate the maximum and minimum results.*

result. The classification using selected features were performed using on 2 classes and then 3 classes referred to in this work as the "2-class problem" and "3-class problem" respectively.

The "2-class problem" was used for finding potential features for the "3-class problem" which was the desired network. This strategy of tackling the problem was motivated, from a visual inspection of samples perspective, by an overlap of data samples between the "average" class with either the "good" or "poor" classes. In addition, the "2-class problem" had the computation speed advantage. It was sensible therefore to leave out the "average" class temporarily during the preliminary prototyping stage.

The linear classifier produced poor results as shown in Figure 7.19. The reason for the poor performance is that the data is non-linear. Thus a linear classification technique was bound to fail. In contrast, the MLP classifier trained by the backpropagation using the gradient descent method produced satisfactory results. There was therefore no need for trying other optimisation strategies as the neural network is not part of the task domain but is only used as a tool.

The interpretation of the results in terms of the confusion matrices and the loss matrices enhanced the quality of the result.

7.4.1 Discussion

The first order statistics (FOS) gave a poor result on paper. The standard deviation from the FOS measures the variation of the data from the mean and since the difference in quality between samples was very small, this feature would give a poor performance. The paper is two-dimensional, consequently, FOS was expected to give a poor performance as it is one dimensional and also as it is not sensitive to the rearrangement of pixels in an image.

The fractal dimension (FD) and the lacunarity features were useful indicated in Figure 7.13 but were rejected because for example, the graininess information of the specific perimeter method (SPM) is *loosely* correlated to the roughness measure of the FD.

Studies have shown the GLDM's performance being either similar or less than that of the SGLDM. Our preliminary experiments in Table 7.6 did confirm this.

The performance in classification of Gabor features in Figure 7.14 and the discrete cosine transform (DCT) in Figure 7.15 was less impressive. These could be wrong techniques for this problem. The DCT and Gabor perform well on images with edges and lines. Since the distribution of information on these images were nearly uniform or random, the poor performance by these techniques is therefore not surprising.

Some of the features were rejected from the selected techniques based on computational time as indicated in Table 7.6 and one example is the *correlation* feature of the SGLDM. Computational cost here is defined as the time taken to get a result from a technique. Thus computation overhead can be used as a comparison tool between techniques. The angular independent NGLDM recommended by other workers, was introduced to overcome the computation speed limitations of the SGLDM. However, the NGLDM was rejected in this work because information on the direction of texture is useful in texture classification in this context. Texture direction is a result of there being a progressive increase of fineness in the texture. The result of the SGLDM indicates that paper texture in this context has a diagonal structure as high classification performance was in the 45 and 135° directions and the results are shown in Table 7.5. This direction information can be useful to the papermaker as it can provide information on the state and or part of the machine that is contributing to the production of low quality paper. Additionally, there has been a dramatic improvement in the speed of computers, to the SGLDM's favour.

The SGLDM entropy gave a good performance despite the samples from different classes hav-

ing small differences between each other. Entropy characterises the randomness in the distribution of grey level values in the image, naturally, it must be capable of enhancing the classification performance on paper as the surface profile of paper in the machine direction (when viewed using angular illumination) is also random. The SGLDM energy is a sum of square of relative frequencies (co-occurrence matrix values) hence its good performance on paper.

The short run and the long run emphasis features of the GLRLM were expected to give good results as they measure the degree of coarseness of the image. A course image is characterised by large runs whereas a fine image is characterised by short runs. The classical GLRLM features characterisation of the surface appearance for quality was satisfactory and hence it was not profitable to use the new GLRLM features of [53][54] used by Tang [55]. The new Short run high grey-level emphasis (SRHGE) and long run low grey level emphasis are given by: $SRHGE = \frac{SRE}{j^2}$, $LRLGE = \frac{LRE}{i^2}$ Thus the only difference between the new GLRLM and the classical GLRLM are in the denominators of these relations. The reason why there was no significant difference in te results is because there might have not been variations between the distribution of grey level and the distribution of the grey level run lengths.

The specific perimeter method (SPM) is based on the perimeter pixels of blobs which characterise the random nature of surfaces. Since paper samples were captured in the machine direction (MD) which is random, the modified SPM was expected to yield better classification performance. Its under-performance in this work could be due to the overlap in the average class with either of the 2 classes.

The results from the PCA were not impressive and the reason could be because the data is non-linear yet the PCA is a linear technique.

The results were evaluated using confusion matrices. The ideal result has a confusion matrix with values only on the leading diagonal. The confusion matrix in Table 7.7 shows that errors occur due to misclassification of paper on class boundaries, consequently, errors due to "good" samples being classified as "poor" are less costly. This error could be due to the overlap of sample classes.

The backpropagation which uses different random initialisations which produce different weight sets is a potential source of noise. However, this handicap was compensated through multiple training runs (10 runs). A larger training set was used as it allows the network to refine the knowledge [136] and errors are reduced after the addition of each batch of the training patterns.

Whereas the human expert cannot be guaranteed to be 100% perfect in classifying the samples, its contribution in this respect is recognisable and the loss is insignificant.

Overall, the less than optimal performance is attributed to the close similarity between the "average" class and either of the two classes.

7.5 Further work

In terms of future work, combining features or classifiers using data fusion might provide a solution. In the section that follows, we show how these schemes could be applied in paper making. Furthermore, instead of using the mse, a training strategy with a function that asks real cost must be used.

7.5.1 Data fusion

Data fusion[138] is the combination of data from different sources in order to improve classification. The acquisition, processing and combination of information provided by different knowledge sources is referred to as multi-sensor data fusion. Data fusion exploits the strength of individual techniques.

Data fusion methods are categorised under centralised fusion, hybrid fusion and decision level fusion amongst others. This section covers a few data fusion strategies and their potential in improving paper classification.

7.5.2 Introduction

Data fusion[138] is a method used for combining data. When used with classifiers, it is motivated by the assumption that different classifiers offer complimentary information so that the error made by one classifier about the patterns to be classified [134] is rectified by the other classifiers. Each different classifier therefore covers a partial part of the solution space [139] leading to a better a posteriori probability (decision). Consequently, classifiers which individually perform poorly can also be useful when combined with others. The other motivating fact is the humans' ability of combining multiple senses and experience in tackling different situations.

In terms of texture classification, the data fusion at both feature level and decision level can be useful. The decision level fusion comprises the weighted decision methods, dempter-shafter method, the Bayesian inference and classical inference. This thesis will not discuss any of these data fusion strategies as the aim is not to compare but to highlight the potential usefulness of data fusion to this work.

An example is a neural-based data fusion [134] which combines outputs from different neural networks that are trained on the same data set forming what is called a committee. The generalisation performance becomes the average of the classification performance of each of the best networks. Bishop [134] has found that a committee can outperform a single classifier.

Data fusion based on learning reduces correlation errors amongst classifiers. For example, in classification a pattern is assigned to one of the given classes. Each classifier w_k is modelled by the probability density function $P(x_i|w_k)$. The classifier's prior probability of occurrence is denoted by $P(w_k)$. The following are some of the data fusion strategies.

7.5.3 The Median Rule

$$P(w_i|xi) = max_{k=1}^{m} medP(w_k|x_i) \tag{7.4}$$

The combined decision is the median of the a posteriori probabilities. xi is a pattern vector.

7.5.4 The Sum rule

Assuming equal prior, the sum rule computes the average a posteriori probability for each class over all classifier outputs.

$$\frac{1}{R}\sum P(W_j|x_i) = max_{k=1}^{m}\frac{1}{R}\sum P(w_k|x_i) \tag{7.5}$$

where R is the total number of classifiers. A pattern is assigned to a class with a maximum average a posteriori probability. However, if one of the classifiers' output is an outlier, the average will be extremely low or high and hence an incorrect decision might result. The assumption that the posterior class probabilities do not deviate greatly from the priors might be incorrect for some applications. However, comparative studies by Kitter et al [138] show that the sum rule is the best classifier combination scheme and it is most resilient to estimation errors.

7.5.5 Product rule

The Product rule [138] assumes conditional independence and as a consequence combines the a posteriori probabilities generated by individual classifiers using a product. It provides an adequate approximation to a solution. It performs well even where the conditional independence assumption does not hold. However, a single classifier can inhibit the result if its output is close to zero probability. This impacts badly on a decision rule combination as all classifiers in the worst case might provide their respective decisions for a hypothesised class identity to be accepted or rejected.

7.5.6 Fuzzy fusion

The implementation of the fuzzy system involves the extraction of linguistic classification rules from real-valued examples. In a fuzzy system [138] the OR and AND are key operators. In the case of combining classifiers, the input to the fuzzy fusion system are the outputs from different classifiers. In the OR a candidate is accepted as a sample if at least one classifier produces a result greater than the corresponding threshold. In the AND all the classifier outputs must be greater than the corresponding thresholds. Thus Fuzzy clustering [140] can combine outputs from different classifiers and the benefit is that the fuzzy output has a degree of quality attached to it. Additionally, it can handle missing and inaccurate data and is useful where data are uncertain, real-valued and multi-dimensional.

However, the fuzzy system suffers from the explosion of linguistic rules which results in high dimensionality and high computational complexity. Furthermore, in extracting these rules an important rule might be missed. It might therefore not be profitable to use this fusion strategy for paper classification.

A practical application of data fusion is in the design of a person authentification system [141] which is evaluated by using the false acceptance rate (FAr) and the false rejection rate (FRr). FRr is the rate of false rejections over the total number of authentication tests. Several pairs of FAr and FRr corresponding to different operating points of the fusion system are used.

7.5.7 Data Fusion Summary

The fusing of data from multiple classifiers has a potential for reducing uncertainty, improving resolution, system reliability, robustness and also enhancing the performance of a texture classification system. Furthermore, data fusion should lead to higher classification performance compared to each classifier used individually. An alternative scheme would be the papnet system [142] discussed in the next section. For the product rule, a single classifier with an infinitesimally small output can inhibit the result and hence classifiers in the worst case might provide their respective decisions. In the case of the sum rule, an outlier classifier output can lead to an incorrect decision. However, comparative studies by Kitter et al [138] show that the sum rule is the best classifier combination scheme and it is most resilient to estimation errors. The fuzzy system suffers from the explosion of linguistic rules, consequently a high dimensionality and high computational complexity results.

There is similarity between data fusion and the method used in thesis where features from different techniques were combined in some way. Consequently, data fusion can be useful in characterising the paper surface appearance. The Median rule data fusion has a potential of giving a more useful result than the other data fusion strategies discussed in this thesis when used for characterising the surface appearance of paper. This technique is intuitively appealing as a median potentially eliminates the extremes of results (an exceptionally high result and an exceptionally low result).

7.5.8 Papnet System

Mark Rutenberg adapted the NASA digital image processor used on images from Mars to automate the analysis of smear tests. The Papnet [142] detects abnormal cells categorised as negative. It is a blind re-screening procedure. PAPNET consists of two independent neural networks for screening abnormal cell clusters and single abnormal cells respectively. This screening is performed irrespective of cell overlapping, shape or staining. Each field of view (FOV) is given a score related to the closeness of the networks' outputs to their maximum or minimum. Inbuilt in a neural network is a wide variety of normal and abnormal cells. In terms of manufactured paper, this system can be used to classify only "good" paper and then the "average" and the "poor" classes can be left to be classified by other means.

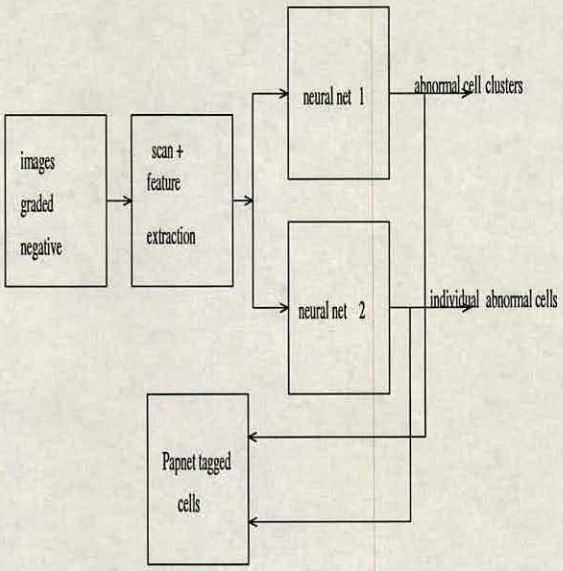


Figure 7.20: *This a Papnet system. Negative means the samples that are abnormal.*

7.5.9 Novel Detector

The novel detector is built out of "good" samples and it detects "poor" samples as novel and also inputs taken from regions with very low training data density[139]. A classification result is therefore only valid if the input comes from a class well-represented in the training set. The network extrapolates for very low data input, consequently, its output is not a statistical inference based on training data but a consequence of any prior information. Extrapolation is typical when the network is tested on a set of conditions different from the training conditions. There is a striking similarity of this method and that of the papnet system. In terms of paper classification, this approach can be used to screen "poor" paper leaving behind the "average" and the "good" paper to be classified by other means. The "average", from the earlier discussion of the results, it was felt that it carries a lesser risk in terms of customer satisfaction compared to the "poor" sample. This approach is useful since it turned out that the number of poor quality paper was low and it is the intention of the papermaker to make it even lower resulting in zero defect.

Chapter 8
Conclusion

8.1 Thesis Summary

In chapter 2 it was established that texture is a neighbourhood property that contains enough information which if extracted using appropriate texture analysis techniques could lead to high texture classification performance. In addition, the median filter and the high pass filter were found to be adequate for removing noise from the image. The results on the practical application of these filters on paper were presented in chapter 7. The masks for these filtering strategies were optimised such that they became sensitive to the feature sizes of interest.

In chapter 3, the study of the spatial texture analysis techniques resulted in some of the features being selected for use in characterising the surface appearance of paper for quality and their results are included in chapter 7.

The deduction from chapter 4 was that feature extraction techniques that extract both frequency and spatial information might be useful in the characterisation of the surface appearance of paper. These techniques were abandoned either because it was expensive to tune them so that they became sensitive to the feature sizes of interest or the features were just not relevant to the problem. Some of the features from these techniques provided information that was strongly correlated to the information already provided by features from other techniques. An example is the Discrete Fourier Transform which excels on periodic structures. Since the investigation was based on the machine direction which is random, the DFT would not be of much use on this texture.

In chapter 5 the fractal dimension and the lacunarity features were rejected because, for example fractal dimension provided the roughness information which is loosely correlated to the granularity of the specific perimeter method. The specific perimeter method, also covered in this chapter individually gave a satisfactory performance and when combined with features from other techniques it improved the result.

In chapter 6, automatic feature selection techniques were rejected in favour of feature selection

by hand (manual approach) because the latter has the added advantage of incorporating the knowledge about the features. Where speed is of essence, the manual approach is not favoured. The only automatic feature selection technique used was the PCA. The other feature selection techniques were rejected because they were found to be either, weak, computationally expensive or being suboptimal when used to characterise the surface appearance of paper. Whilst the manual approach is not computationally cheap, it does incorporate expert knowledge. This can be a usable tool since as the user does not have to rebuild it. The background information on Neural Networks (NN) was also covered in this chapter.

In chapter 7 the experiment for optimising the image capture parameters was carried out. In addition, optimal features for the 2 class classifier were computed. The results from features selected from different techniques which were used individually and in combination were presented in this chapter. The optimal features used on the 2 class problem were later used on the 3 class problem. The interpretation of the results using the confusion matrices and the loss matrices helped identify the problem of the low classification error due to the unbalanced set.

Classification was given as the average of 10 runs which is fine for the mean. The averages and the min-max as intervals were used as measures for statistical significance. The satisfactory performance of the PCA compared to our intuitive selection of features was expected since PCA is a linear technique, as a consequence, it struggles where there is overlap between different classes of data. What makes the PCA unique is that it rotates the axis in search for a "trough" that separates the classes, in this case there was no change in the result.

The linear classifier results were very poor. The reason just like in the case of the PCA, it excels on linear data yet this data is non-linear. The results from an MLP classifier were impressive because the data is non-linear and that the MLP is a non-linear classification tool.

8.2 Summary Conclusions

The characterisation of the surface appearance of paper for quality has been addressed. The features from texture analysis techniques combined in some way are capable of discriminating different classes of paper samples.

8.2.1 Introduction for the Conclusion

The thesis has explored the use of texture analysis techniques to find out if the surfaces of samples of paper can be classified from texture. It turns out that combining the features from different texture analysis techniques gives a good result. These techniques included the SGLDM, the GLRLM, the FOS and the SPM. Further work should extend this work by using data fusion strategies [138] to combine results from different classifiers trained from features from these techniques.

8.2.2 Detailed Conclusions

The use of multiple features from different techniques in combination gave a better coverage of discriminative information than when only features from a single technique are used. The use of multiple features was motivated by the realisation that the characterisation of paper was difficult and that each feature might capture a complimentary and desired portion of information.

Features that were useful in characterising the surface appearance of paper were from the SGLDM, the FOS, the GLRLM and the SPM techniques. The SGLDM is based on grey level transition or spatial inter-pixel relationships [45]. Since the surface is two dimensional the SGLDM must be suited to analysing surfaces of textured materials. Its performance on paper was satisfactory. The paper seemed to have a diagonal structure as the best results from the SGLDM were in the 135 and 45° direction. On preliminary work, a combination of SGLDM features computed from different inter-pixel displacements did not improve the result.

The SGLDM normally performs better than the GLRLM technique on related problems. However, on paper it performed worse than the latter possibly because of the fuzzy boundary between the "average" class and either of the 2 classes. As a consequence, the transition from one grey level to its neighbour(s) is very small if not none. This certainly does not favour a technique that flourishes on differences in neighbourhood pixels. The SGLDM is ineffective in characterising relatively random, low contrast textures [59] and hence its high generalisation error compared to GLRLM on paper. The GLRLM can potentially give better results in an environment where neighbouring pixels have or almost have the same grey level value. It is unlikely that median filtering could have affected the SGLDM result as the median filter preserves edges in the image and in this context grey level transitions.

The study of the SPM was motivated by the interest shown on it by the paper industry due to

previous successes in the measurement of the formation of paper covered in chapters 1 and 5. The modified SPM's performance alone in characterising the surface appearance of paper was less than expected. However, in combination with features from techniques mentioned in the previous paragraphs, it improved the result. The SPM was recently used by Lee et al [143] in characterising the surface appearance of coated paper.

The visual difference in surface appearance between samples in different classes was very small. As a consequence, the linear classifier performed badly on this data as shown in the results in chapter 7. The non-linearity of data was evident in the preliminary work we carried out using scatter plots of features and the data was not separable. It turned out that the multilayer perceptron (MLP) neural network which is a non-linear classifier, produces a good result. The neural network is not new, but the application of features from texture and neural networks together in characterising the surface of coated paper is new. The use of texture features (multiple features) from different techniques in combination to characterise the surface appearance of paper is novel.

There could be a classifier whose performance is better than that of an MLP, but the MLP has been found to be a reliable classification tool over the years. Thus there was a saving in time by using an off-the-shelf MLP classifier and the benefit more energy was devoted on identifying features from texture that can adequately characterise the surface of paper.

Previous investigations on paper quality assessment focussed on characterising formation [92]. Formation is the assessment of the mass variation and the optical densities of the paper. Trepanier used an image analysis system that incorporates SPM and managed to classify paper. He was assessing paper formation quality. The SPM was found to increase with the formation quality [93]. Paprican has developed software that employs the SPM to evaluate formation quality [92]. SPM has been used recently in [143] for characterising the surface appearance of coated paper. The SGLDM and SPM are therefore not new to paper quality assessment but the application of the modified SPM and SGLDM on surface appearance of paper and not its formation is new.

The selection and the combining of features formed an important component of the work. Computer image analysis is now entrenched in industry although it struggles to achieve the same in paper manufacturing. Experimental results in chapter 7 indicate that this scheme is feasible, is computationally less complex and is better than previous work on paper. There is scope for

improvement and suggestions on how this can be tackled have been included in this thesis.

The classification of paper is a difficult problem as revealed by the results from confusion matrices in chapter 7.

We now catalogue the successes scored by this thesis: The use of a loss matrix to interpret classification results of paper is new. The loss matrix has been used to calibrate classification error in terms of cost. A combination of complementary features from different techniques performed better than features from a single technique. A system that "learns" the subjective judgement of the operator and then provides classification of paper has been developed. This thesis has successfully built a vision system that matches the capabilities of the human element in assessing the quality of the surface appearance of paper.

8.2.3 Conclusions for operational use

At best, this thesis achieved 87% correct classification on the "poor" and "good" paper samples. It also achieves repeatability in the measurement of quality of the surface appearance of paper and an acceptably accurate classification performance. This system, based on texture analysis approaches the performance of the human operator and is attained at computationally realistic costs, for example, testing takes less than 30 seconds. This vision system is potentially useful to the paper manufacturer as it standardises the assessment of quality.

The quality assessment of the surface appearance of coated paper has been automated. Further work must build knowledge into the system by incorporating data collected at regular intervals from each Mill over a stipulated period of time. In addition, the data from each shift could be assessed separately. The use of this data in training a neural network is likely to enhance the usefulness of the result. The data fusion strategies are an elegant way of combining information and they should be tried on this work.

8.2.4 Global Conclusions

The thesis has examined "the suggestion that an *automated paper-classification system* based upon multiple texture-based features and trained to matches a human operator can approach the performance of the human visual system" given that all were classified when the human performance is at its best, for example, performing classification of samples as a first thing

on arrival at work. The level of classification above 80% is acceptable to the manufacturer otherwise a classification level above 95% will be very useful. This objective has been met partially as a classification of 87% achieved with two classes differs from the human visual system by 13%. This based on the assumption is that the classifier is trying to see what the human expert sees when grading paper. As a consequence, features based on texture extracted from paper have a potential of matching the human visual system. This could be subjective but the combination of the judgement of experts from Tullis Russel who have been in the job for years makes the human visual system (HVS) a useful benchmark in this context. However, it would have been useful to use a number of non-experts as well in classifying the paper to measure HVS performance. The following are the likely reasons why the system differs from the human visual system:

- The classes are too close to one another.
- The classifier is suboptimal as it trains to the stopping criterion. As a consequence, training might terminate before a global minima is reached.
- The gradient descent method used with the MLP is not an optimal training strategy.
- There might be missing data.
- The data collected was small (only 337 different samples).
- The mse used in the classifier is not a good cost function.

This could be further improved by implementing the following:

- by using a better classifier, for example the support vector machines (SVM).
- Improving data collection, that way the fuzzy class will be reduced.
- Using a classifier that uses the optimal conjugate training strategy instead of the gradient descent approach.
- Increase the number of samples collected from 337 to 1000 samples. NB. splitting this data into validation, training and test meant that in real terms the original samples used for training are $\frac{337}{3}$ which might be very small to map a complete decision space. NB: classification rate increases with an increase in the number of samples for each class.

- Using a cost function that interacts with the classifier.
- In addition to using features from different techniques in combination, different classifiers trained using the features that were selected by this thesis can be used in combination.
- The use of both experts and non-experts in assessing the variability of humans might assist in establishing the useful level of performance.

8.2.5 Contribution to Knowledge

This study explored the use of a wide range of features from spatial, spectral and fractal based techniques to assess the surface appearance of paper. This is the first time multiple texture based techniques have been studied on paper. A number of texture analysis techniques that can potentially characterise the surface appearance of paper have been identified. The "intelligent" feature selection technique based on intuition in the context of papermaking has been applied for the first time. We have found that classifying paper is a difficult problem.

The *spatial grey level dependence matrix* and the *specific perimeter method* are not new to paper quality assessment, they have also been used by Lee et al in the characterisation of the surface appearance of coated paper [143]. Their work differs from this work because they used UV light instead of white light. The use of the modified specific perimeter method in characterising the surface appearance of paper is therefore new. The use of features from texture (extracted using different techniques) and the MLP together for classifying the surface of coated paper is new.

The surface of manufactured paper has been successfully characterised using information from texture and this is a milestone in the surface quality measurement for the papermaking industry. The use of confusion matrices and the loss matrices helped to interpret the quality of the result. The loss matrices is used to calibrate the classification performance in terms of cost. This is the first time that these matrices have been used on paper.

In conclusion, the use of multiple texture-based features from different techniques in combination to duplicate the functions of the human operator in assessing the surface quality of paper is new.

Appendix A
Miscellaneous

A.1 Camera

This appendix presents a brief review of the salient features of a camera. The understanding of the basics of the operation are critical to the capture of a good image. The diaphragm of the camera varies the amount of light reaching the film at the lens (exposure) and it also controls the image depth of field. The latter is the degree to which portions for the subject near and far from the camera look sharply focussed and is determined by the lens f - number and the focal length. The shutter speed regulates the amount of light reaching the film.

A wide $(f/2)$ and a small $(f/64)$ opening of a diaphragm give a shallow and a better depth of field respectively. A telephoto lens's image is prone to atmospheric effects and the result is a degraded image quality. Thus a high quality lens is recommended. The ratio of the focal length to the diameter of stop is the F-number given by f/d and it determines the quality of the images. A long focal length enlarges the image size. In this work, a Olympus C-2500L digital camera was used. This camera had an on-camera storage and a removable storage.

The image quality depends on the resolution of the camera which is defined as the number of photosites on the CCD array. The $imagewidth \times pixelswidth$ and $imageheight \times pixelsheight$ is the CCD dimension. The speed, dynamic range, pixel area, square pixel and image size are some of the important parameters of a camera. For example, the sensitivity of the pixel to light is proportional to the area size. Additionally, tiny details can be captured by a high resolution camera.

The camera was focussed on the area to be captured in the way the human visual system does. A milestone in this work was that we could see on the captured image the features that were observed on the sample.

The processing of an image in a domain closer to where vision takes place is important for a successful discrimination of information. The HVS views surfaces with varying resolutions, contrast and brightness settings among other parameters. The way peripheral information is

selected and used is not known. The fovea deals with the detailed image whilst the parafovea deals with neighbouring areas. The other salent parameters considered during image capture have been included and discussed in chapter 7 on a section that deals with image capture.

A.2 Coding

Coding was done in C programming language. The scripts used for example, for combining data samples from different grades for use as inputs to the neural network was written in PERL. This made it easy to select the number of rows of data that was needed at any time as inputs to the classifier especially during feature selection experiments.

Doing ther initial coding for each algorithm using C langauage might be a waste of time as a lot of time might be spent in trying to have the code run and yet after preliminary experimentation it might be found that the algorithm is not useful. Thus it is beneficial to use an off the shelf algorithm if its available for protoyping and in our case we used Matlab and WIT for prototyping. However, WIT is not flexible, for example, some of its algorithms cannot be modified or extended. Thus once an algorithm produced useful results, it was then coded in C language so that flexibility could be incorporated into the code. Additionally, WIT is expensive and not widely available as it is only used by British Aerospace among a few companies. Thus it was sensible to to write the final code in such that it could be taken to our industrial partners for demonstration. Furthermore, many people have used C, thus it would be easy to get someone who can understand the programme even after many years yet WIt could be obsolete by then. Writting own code from mathematical equations was very useful as it enhanced understanding of the underlying concepts of the techniques (feature extraction algorithms). Furthermore, it was also easy to modify the code to suit specific requirements as they arose. Paintshopro was used for data conversion from the TIF format to the pgm format and the conversion can be done in batch process. MATLAB was also used extensively during the prototyping stage. C was then used to write the final programme.

This implementation strategy had an advantage of speed as during prototyping some of the algorithms were found to be useless thus time which would have been used to code a useless programme using C language was saved.

Appendix B
Results

This appendix presents the results from experimentation. Some of these results have been presented as bar charts in chapter 7.

		classified as	
		Good	Average
Actual	Good	33.289	17.456
	poor	13.256	36.000

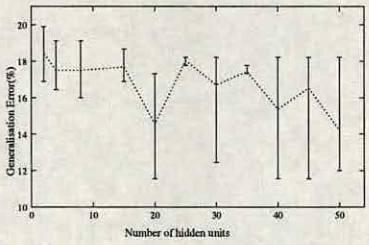
Table B.1: *The table shows the results for the best FOS features, the standard deviation and kurtosis for the 2 - class exp1 in Table7.3*

B.1 The results from the Principal components analysis

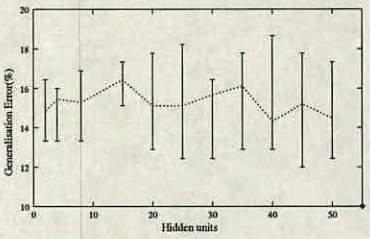
The PCA plots Figure B.3 against the number of hidden units show a trend. The average percentage generalisation error is decreasing with an increase in the number of hidden units. The lowest point for all the PCAs in the 2-class problem was at 30 hidden units. In contrast, the lowest for the 3-class problem also considering the lower dimensionality was at 20 hidden units using 4 and 5 principal components.

B.2 State Probabilities

This appendix presents the state probabilities. The aim is to check by hand what is expected of the neural network's minimum performance. This also helps especially in case where the data is not balanced (unequal data samples in the classes represented in the problem). This is achieved through computing the percentage classification error needed in order to beat the class probabilities. This involves finding out from the training data set which class has the most examples. Are there more good ones, than poor ones? Let's say that most them are in the good class. In the test set count how many good ones there are in total. Assuming that the classifier

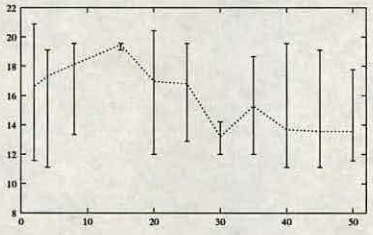


(a) The graph shows classification results from 12 features from features chosen from correlation. They are from the FOS, the glrlm, sgldm and the spm

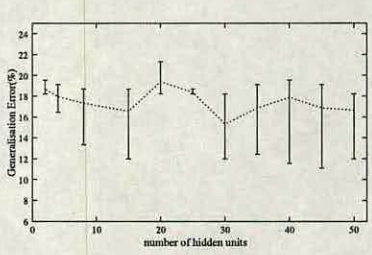


(b) The graph shows classification results *for "Combined 6" from the spm, glrlm, and sgldm* used to discriminate the poor and the good paper class

Figure B.1: *The* "Combined 6"

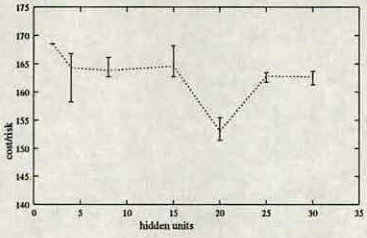


(a) Graph showing classification results for the 2-class problem for surface appearance features extracted using 6 principal components.

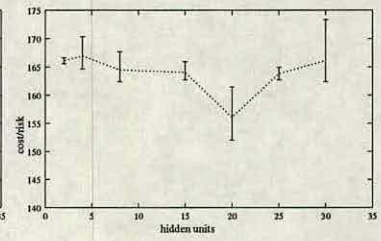


(b) Graph showing classification results for the 2-class problem for surface appearance features extracted using 8 principal components

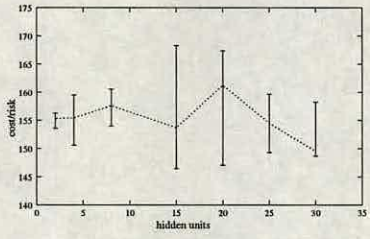
Figure B.2: *The* Principal Component *Analysis. The error bars indicate the maximum and minimum results. The MLP classifier was used*



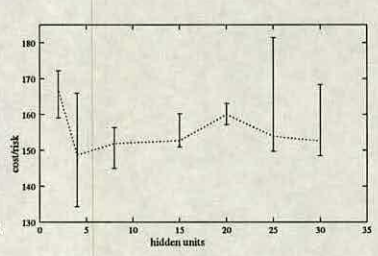
(a) Graph showing classification results for the 3-class problem for surface appearance features extracted using 4 principal components and MLP classifiers. The error bars indicate the maximum and minimum results.



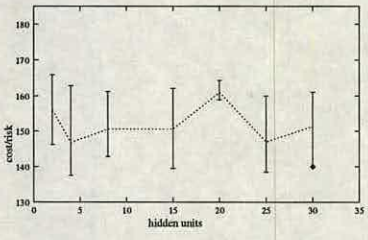
(b) Graph showing classification results for the 3-class problem for surface appearance features extracted using 5 principal components and MLP classifiers. The error bars indicate the maximum and minimum results. package



(c) Graph showing classification results for the 3-class problem for surface appearance features extracted using 6 principal components and MLP classifiers. The error bars indicate the maximum and minimum results.



(d) Graph showing classification results for the 3-class problem for surface appearance features extracted using 8 principal components and MLP classifiers. The error bars indicate the maximum and minimum results.



(e) Graph showing classification results for the 3-class problem for surface appearance features extracted using 9 principal components and MLP classifiers. The error bars indicate the maximum and minimum results.

Figure B.3: *The* Principal Component *Analysis*

will say that all the patterns are good - then we need to compute the percentage error using the following formula:

$percerror = 100 - 100 * (goodpatterns/totalpatterns)$

An an example, if 40% of the test set are good patterns then our neural network must do better than 60% error before its worth using. This, approach at least tells us how much better the features are than just using random numbers.

In the context of our work, the number of Patterns in the training, validation and test sets were 565, 560 and 560 respectively.

This is the Training set:

- No. of good patterns : 120
- No. of average patterns : 320
- No. of poor patterns : 125

This is the Test set:

- No. of good patterns : 100
- No. of medium patterns : 400
- No. of bad patterns : 60

If a pattern is selected at random then the probabilities that it would be good, average or poor are:

- The probability for good class :$120/565 = 0.21$
- The probability for the average class :$320/565 = 0.57$
- The probability for the poor class :$125/565 = 0.22$

These probabilities are called the training set class probabilities.

In the case that the neural network took the easy route and classified all the patterns under the class that has the highest probability, which one would it choose?

The following are the percentages that the classifier will get:

$5/7 = 400/(100 + 400 + 60) = 71.4\%$ if the classifier says that everything is class "average" then it will get 71.4 percent patterns in the test set correct (because there are 400 in the average class and a total number of patterns of 560).

Similarly, it will get $2/7 = 160/(560) = 28.6\%$ patterns the ones that it gets wrong by saying that everything is a "average" Thus the neural network will have to do better than a percentage error of 28.6% before it beats just using the class probabilities.

B.3 Statistical significance

This is an overview of procedures followed in evaluating results in terms of statistical significance. This is achieved through the computation of 95% CI of means. A good probability of rejecting any true null hypothesis (Ho) is achieved through choosing a threshold. The key is knowing whether some of the classifier results which are statistically significant are wrong.

B.3.1 The p-value

A $p - value$ of a test result is the amount of evidence against (Ho). It is useful where either (Ho) is rejected or where it cannot be rejected. The p-value is the probability of observing a test statistic at least as extreme as the value actually observed. It is a measure of support that $\mu \neq \mu_0$. It measures the strength of the results of a test, in contrast to a simple reject or do not reject options. The p-value is combined with the significance level η (threshold) to make a decision on a given test of hypothesis. The p-value is defined with respect to a distribution and it is called a "model-distributional hypothesis".

The p-value represents the probability of making a TYPE I error (rejecting the Ho when it is true). The interpretation of different values of θ:

- $P < 0.01$ very strong evidence against H0.
- $0.01 P < 0.05$ moderate evidence against H0.
- $0.05 P < 0.10$ suggestive evidence against H0 (lower support).
- 0.10 P little or no real evidence against H0.

When a $p - value$ is associated with a set of data, it is a measure of the probability that the data could have arisen as a random sample from some population described by the statistical model. The distribution of p-values under Ho is uniform, and it is thus independent of a particular form of the statistical test.

In summary, if Ho is true and the chance of random variation is the only reason for sample differences, then the p-value must be included in the decision making process as evidence. For the fixed-sample size, when the number of realisations is decided prior, the distribution of p is uniform (assuming the Ho). The means that are far part is evidence that the samples come from different populations. Then the Ho at $p = 0.05$ can be rejected.

B.4 Confidence interval

This appendix presents the Confidence Interval (CI) and its key function is to inform the investigator on how confident he can be on the result. CI is used to interpret the given result. CI is a range of values calculated from a sample. The 95% CI contains some population from which the unknown true mean is being estimated. The CI does not assess the significance.

The 95% is called the confidence level γ (given by $(1 - \alpha)$) for the confidence interval (CI) and it specifies the success rate of the method used to construct the interval. The implementation of CI involves choosing a γ and then determining the corresponding c. The mean of the sample is computed. It is viewed as the approximate value for which a parameter can at most deviate from the unknown true value. The wider the CI the higher the γ. The CI levels used are 95, 99, 99.9 and 90. Thus in practice, from CIs hard conclusions cannot be drawn from small differences between recognition scores. Then $k = c\sigma\sqrt{n}$.

$$CONF\theta_1 \leq \theta \leq \theta_2 \tag{B.1}$$

The upper and lower confidence limits θ_1 and θ_2 are the interval estimate for the unknown parameter θ. γ is the probability of getting an interval that will include the unknown exact value of the parameter.

CI width of a population mean $\mu = 2z_{\frac{\alpha}{2}}\sigma\sqrt{n}$
We want high confidence levels and narrow CI. Our control is the CI width.

As confidence level increases $(1-\alpha)100\%$, the width of CI increases.

As variance increase, width of CI..... increases

As sample size n increases,the width of CI the width of the CI decreases.

The value of the sample does not affect the CI width.

Sample size required to estimate μ to within B with $100(1-\alpha)$ confidence.

$$\frac{n = (z_{\frac{\alpha}{2}}\sigma)}{B} \tag{B.2}$$

B.5 Statistical evaluation

This appendix presents the analysis of variance (ANOVA) used for interpreting (or analyse) results and not improve any result. The Ho for ANOVA is that the classes come from one population (they have the same mean). Different means can only arise as a result of sampling error (samples differ by chance).

Another parameter, the variance ratio F is 1 if the 2 classes come from the same population. Large F is evidence that the classes of paper differ other than just by chance. If $F_{table} > F_{computed}$, then there is no evidence to reject the hypothesis. If $F = 0$, the feature explains nothing. What makes F useful is that the mean might be the same but the spread might have changed.

One way ANOVA tests the differences in mean for many classes. In terms of manufactured paper, Ho is $\mu_{good} = \mu_{average} = \mu_{poor}$. All the time the evidence is from p-value for rejecting 3 means. If the hypothesis is rejected, then the 3 classes are not equal, which implies that H1 is accepted. It might happen that $\mu_{good} = \mu_{average}$ or $\mu_{average} = \mu_{poor}$ or $\mu_{good} = \mu_{poor}$.

Then a simple 2-Sample T- test which tests two classes at a time for Ho can be carried out. Ho is tested by observing the means of 2 samples and the variability around each of them. The degree of confidence must be traded off against usefulness, otherwise CIs will be exremely wide and useless. The Two-Sample t and a test of significance determine whether sample 1 or

sample 2 population means differ significantly at the 0.1 level. We are 95% confident that the "good" grade data lies within $(0.05675, 0.06197)$. The p-value for 2 means are an evidence for rejecting the hypothesis. The following are the thresholds chosen when applying these tests:

- $P < 0.01$ very strong evidence against H0.
- $0.01P < 0.05$ moderate evidence against H0.
- $0.05P < 0.10$ suggestive evidence against H0 (lower support).
- 0.10 P little or no real evidence against H0.
- $If sample > 30$, then use Z test. It tests 2 population means.
- $If sample < 30$, use the simple 2 Sample T-test.
- For a significance level, 5% or 1% or both can be used.

This method was not used in this thesis because it was felt that the confusion matrices combined with the loss matrices could do equally well in intepretating results. However, in terms of further work it will be interesting to find out how this enhances the quality of the intepretation of the results.

B.6 Publications

1. Barnabas Ndlovu Gatsheni et al, "Automatic Classification of Surface Appearance Quality in Papermaking using Texture analysis" has been submitted to *TAPPI JOURNAL.*

References

[1] R. Chin and C. Harlow, "Automated visual inspection: A survey," in *Transactions on Pattern Analysis and Machine Intelligence*, vol. 4, pp. 557–567, IEEE, November 1999.

[2] I. Mackay, "Personal communication from ian mackay of tullis russell," Tullis Russell & Co. Ltd, July 2001.

[3] D. Harris, "The nature of industrial inspection," in *Human factors*, no. 2, pp. 139–148, 1969.

[4] R. Day, "Visual spatial illusions: A general explanation," in *Science*, vol. 175, pp. 1335–1340, March 1972.

[5] S. Coren and J. Girgus, "Visual spatial illusions, many explanations," in *Science*, vol. 179, pp. 503–504.

[6] J. Schoonard and J. Gould, "Field of view and target uncertainty in visual search and inspection,"

[7] V. L. Gool, P. Dewaele, and A. Oosterlink, "Texture analysis and ano," in *Computer Vision, Graphics and Image processing*, vol. 29, pp. 336–357, September 1983.

[8] R. Conners and C. Harlow, "A theoretical comparison of texture algorithms," in *Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, pp. 204–222, IEEE, May 1980.

[9] B. Julesz, "Texture and visual perception," in *Transactions on Information Theory*, vol. IT-8, pp. 84–92, IRE, February 1962.

[10] M. H. Davis M.N., Roehr W.W., "An instrument for formation measurement," in *Paper Trade Journal*, vol. 101, pp. 43–48, July 1935.

[11] S. Weszka, C. Dyer, and A. Rosenfeld, "A comparative study of texture measure for terrain classification," in *Transcations on systems, man, and cybernetics*, vol. SMC-6, pp. 269–285, IEEE, April 1976.

[12] R. Conners, C. McMillin, K. Lin, and R. Vasquez-Espinosa., "Identifying and locating surface defects in wood: Part of an automated lumber processing system," in *Transactions on Pattern analysis and Machine intelligence*, vol. 5, pp. 573–582, IEEE, 1983.

[13] H. R. Siew L.H and W. E.J., "Texture measures for carpet wear assessment," in *Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, pp. 92–105, IEEE, Jan 1988.

[14] H.-S. Don, K.-S. Fu, C. Liu, and W.-C. Lin, "Metal surface inspection using image processing techniques," in *Transactions on Systems, Man and Cybernetics*, vol. SMC-14, pp. 139–146, IEEE.

[15] A. Lobo, "Image segmentation and discriminant analysis for identification of land cover units in ecology," in *Transactions on Geoscience and Remote Sensing*, vol. 35, pp. 1137–1145, September 1997.

[16] C. Thierry and L. Philip, "The characterisation of paper formation," in *Tappi Journal*, pp. 175–185, December 1990.

[17] F. Cohen, Z. Fan, and S. Attali, "Automatic inspection of textile fabrics using textural models," in *Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, pp. 803–808, IEEE, August 1991.

[18] L. Macaire, V. Ultre, and J.-G. Postaire, "Real-time control of galvanized coating aspect by a texture inspection system," in *Systems, Man and Cybernetics, International Conference on Systems Engineering in the Service of Humans', Conference Proceedings*, vol. 2, pp. 493 –498, IEEE, October 1993.

[19] R. Trepanier, "Off-line paper formation quality testing and its dependency on forming technology," in *Journal of Pulp and Paper Science Vol No. 1987*, vol. 13, pp. 111–115, July.

[20] G. Burkhard, P. Wrist, and G. Mounce, "Pulp paper," in *Pulp Paper Magazine, Canada*, p. 139.

[21] O. Kallmes and J. A. Ayer, "Light scanning system provides qualitative formation measurement," in *Pulp and paper*, pp. 99–105, April 1987.

[22] S. Kapoor and S. Wu, "A stochastic approach to paper surface characterisation and printability criteria," in *J Phys. D:Appl. Phys.,*, vol. 11, pp. 83–96, 1978.

[23] C. Thierry, H.Tomimasu, and L. Philip, "Characterisation of paper formation," in *Tappi Journal*, pp. 153–159, July 1990.

[24] M. Demeyer, *MSc. Thesis*. SUNY College of Environmental Science and Forestry, Syracuse, N.Y., 1982.

[25] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis and Machine Vision*. Chapman and Hall.

[26] M. Chantler and G. McGunnigle, "The response of texture features to illuminant rotation," in *Proceedings. 15th International Conference on Pattern Recognition*, vol. 3, pp. 943–946, September 2000.

[27] S. Jae, *Two dimensional signal and image processing*. Prentice Hall International, Inc.

[28] R. Jain, R. Kasturi, and B. Schunck, *Machine Vision*. McGRAW-HILL INTERNATIONAL EDITIONS, 1995.

[29] A. Kulkarni, *Artificial Image understanding*. Van Nostrand Renhold, 1994.

[30] R. C.Gonzalez and P. Wintz, *Digital Image Processing*. Addison-Wesley Publishing Company.

[31] D. Gabor, "Theory of communications," in *Journal of Inst Electr Eng*, vol. 93, pp. 429–457, 1946.

[32] D. Marr, *Vision - A computational Investigation into the Human Representation and Processing of Visual Information*. H.W. freeman and Co., San Fransisco, 1982.

[33] A. Jain, *Fundamentals of digital Image Processing*. Prentice-Hall, Eaglewood Cliffs, NJ, 1989.

[34] L. X. Wang and J. M. Mendel, "Fuzzy adaptive filters, with application to nonlinear channel equalization," in *Transactions on Fuzzy Systems*, vol. 1, pp. 161–70, IEEE, August 1993.

[35] L. Hertz and R. Schafer, "Multilevel thresholding using edge matching," vol. 44, pp. 279–295, Computer Vision, Graph and Image processing 1988.

[36] C. Chow and T. Kaneko, *Boundary detection of radiographic images by a thresholding method.* Academic Press, New York.

[37] J. Weszka, "A survey of threshold selection techniques," in *Computer Vision ,Graphics and Image processing*, vol. 7, pp. 250–265, 1978.

[38] P. Burt, "fast algorithms for estimating local image properties," vol. 21, pp. 368–382, Computer Vision Graphics Image Processing, 1983.

[39] J. Weszka, R. Nagel, and A. Rosenfeld, "A threshold selection technique," in *IEEE Transactions Comput.*, vol. 23, pp. 1322–1326, 1974.

[40] J. Weszka and A. Rosenfeld, "Histogram modification for threshold selection," in *Trans. on Systems, Man, and Cybernetics SMC-9*, vol. 9, pp. 38–52, IEEE, January 1979.

[41] P. Sahoo, S. Soltani, A. Wong, and Y. Chen, "A survey of thresholding techniques," in *Computer Vision, Graphics and Image processing*, vol. 14, pp. 233–260, 1988.

[42] R. Kohler, "A segmentation system based on thresholding," in *Computer Vision, Graphics and Image processing*, vol. 15, pp. 319–338.

[43] S. Wang and R. Haralick, "Automatic multi threshold selection," in *Computer Vision, Graphics and image processing*, vol. 25, pp. 46–67.

[44] R. M. Haralick, "Statistical and structural approaches to texture," in *Proceedings of the*, vol. 67, pp. 786–804, IEEE, September 1979.

[45] R. M. Haralick, K. Shanmugan, and I. Dinstein, "Textural features for image classification," in *Trans. Syst. Man. Cybernetics*, vol. SMC-3, pp. 610–621, IEEE, 1973.

[46] M. Amadasun and R. King, "Textural features corresponding to textural properties," in *Transactions on Man and Cybernetics*, vol. 19, pp. 1264–1274, IEEE, Sept 1989.

[47] S. Karkanis, G. Magaolas, M. Grigoriadou, and M. Schurr, "Detecting abnormalities in colonoscopic images by textural description and neural networks,"

[48] S. Karkanis, D. Iakovidis, D. Maroulis, G.D.Magoulas, and N. Theofanous, "Tumor recognition in endoscopic video images using artificial neural network architectures," vol. 2, pp. 423–429.

[49] F. Cohen, Z. Fan, and S. Attali, "Automated inspection of textile fabrics using texture models," vol. 13.

[50] K. Song, J. Kittler, and M. Petrou, "Defect detection in random colour textures," in *Image Vision Computing*, vol. 14, pp. 667–683, 1996.

[51] T. E. Southard and K. A. Southard, "Detection of simulated osteoporosis in maxillae using radiographic texture analysis," in *Transcations on Biological Engineering*, vol. 43, pp. 123–131, IEEE, February 1996.

[52] M. Galloway, "Texture analysis using gray level run lengths," in *Computer Graph. Image Processing*, vol. 4, pp. 172–179, 1975.

[53] A. CHU, C. Sehgal, and J. Greenleaf, "Use of gray value distribution of run lengths for texture analysis," vol. 11, pp. 415–420, June 1990.

[54] B. Dasarathy and E. Holder, "Image characterizations based on joint gray level-run length distributions," vol. 12, pp. 497–502, August 1990.

[55] X. Tang, "Texture information in run-length matrices," in *Transactions on Image processing*, vol. 7, pp. 1602–1609, November 1998.

[56] P. Arul, V. Amin, and D. carlson, "Characterisation of beef muscle tissue using texture analysis of ultrasonic images,"

[57] V. Amin, D. Wilson, R. Roberts, and G. Rouse, "Tissue characterisation for beef grading using texture analysis of ultra sound images," in *Ultrasonics Symposium*, pp. 969–972, IEEE, 1993.

[58] H. Loh, J. Leu, and R. Luo, "The analysis of natural textures using run length features," in *Transactions on industrial electronics*, vol. 35, IEEE, May 1988.

[59] V. Manian and R. Va'squez, "A framework for sar image classification:comparison of co-occurrence and gabor based method," in *IEEE 0-7803-3836-7/97*, pp. 335–337, July 1997.

[60] R. W. C. M. M. Trivedi and C. A. Harlow, "Segmentation of a high resolution urban scene using texture operators," in *CVGIP*, vol. 25, pp. 273–310, March 1984.

[61] D. Marceau, "Evaluation of grey-level co-occurrence matrix method for land-cover classification using spot imagery," in *Transactions on GeoScience and Remote sensing*, vol. 28, pp. 513–519, IEEE, July 1990.

[62] O. Tobias, R. Seara, F. Soares, and J. B. ., "Automatic visual inspection using the co-occurrence approach," in *Midwest Symposium on Circuits and Systems, Rio de Janeiro*, pp. 154–157, IEEE, August 1995.

[63] P. Kyriacou, D. Koutsouris, P. Zoumpoulis, and I. Theotokas, "Computer assisted characterisation of liver tissue using image texture analysis techniques on b-scan images," in *Proceedings - 19th International Conference -IEEE/EMBS Chicago, IL. USA*, pp. 806–809, October 1997.

[64] C.-M. Wu, Y.-C. Chen, and K.-S. Hsieh, "Texture features for classification of ultrasonic liver images," in *Transactions on Medical Imaging*, vol. 11, pp. 141–152, IEEE, June 1992.

[65] F. Argenti, L. Alparone, and G. Benelli, "Fast algorithms for texture analysis using co-occurrence matrices," in *Proceedings Radar and Signal Processing*, vol. 137, pp. 443–448, IEEE, December 1990.

[66] C. Gotlieb. and H.E.Kreyszig, "Texture descriptors based on co-occurrence matrices," in *Computer vision graphics and Image processing*, vol. 51, pp. 70–86, 1990.

[67] J. Weszka, A. Rosenfeld, E. Carton, R. Kirby, and J. Mohr, "A comaparative study of texture measures for terrain classification," in *Computer Vision Laboratory, Computer Science Centre*, vol. TR-361, University of Maryland, March 1975.

[68] M. Augusteijn, L. E. Clemens, and K. A. Shaw, "Performance evaluation of texture measures for ground cover identification in satellite image by means of neural network classifier," in *Trans on GeoScience and Remote Sensing*, vol. 33, pp. 616–619, IEEE, 1995.

[69] Z. Zalevsky, I. Ouzieli, and D. Mendlovic, "Wavelet-transform-based composite filters for invariant pattern recognition," in *Applied optics*, vol. 35, pp. 3141–3147, June 1996.

[70] L. Hoffer, F. Francini, B. Tiribilli, and G. Longobardi, "Neural networks for the optical recognition of defects in cloth," in *Optical Engineering*, vol. 35, pp. 3183–3190.

[71] C. Ciamberlini, F. Francini, G. Longobardi, P. Sansoni, and B. Tiribilli, "Defect detection in textured materials by optical filtering with structured detectors and self-adaptable masks," in *Optical Engineering*, vol. 35, pp. 838–844, Society of Photo-Optical Instrumentation Engineers, March 1996.

[72] B. Hubbard, *The world according to wavelets*. A. K. Peters, Wellesley, Massachusetts.

[73] M. Colestock, "wavelets-a new tool for signal processing analysis," in *0-7803-1343-7/93*, pp. 54–59, IEEE, July 1993.

[74] J. Daugman, "An information-theoretic view of analog representation in striate cortex," in *Comp. Neurosc.*, pp. 403– 424, 1990.

[75] S. Mallat, "A theory for multiresolution signal decomposition," in *Transactions on Pattern analysis and machine vision*, IEEE.

[76] S. Hamidi and R. Adhami, "Wavelet transform for image data compression," in *Proceedings of the 26th Southeastern Symposium on*, pp. 462 –465, 1994.

[77] S. Livens, P. Scheunders, G. van de Wouwer, and D. V. Dyck, "Wavelets for texture analysis, an overview," in *Image Processing and Its Applications, 1997., Sixth International Conference on*, vol. .2, pp. 581–585, IEE, July 1997.

[78]

[79] R. Brooks and S. Iyengar, *Multi-Sensor Fusion, Fundamentals and Applications with Software*. Prentice Hall PTR, 1998.

[80] B. Richard and L. Zhi-Quiang, "Feature extraction and analysis of handwritten words in grey-scale using gabor filters," in *0-8186-6950-0/94*, pp. 164–168, IEEE, 1994.

[81] A. Jain and F. Farrokhnia, "Unsupervised textures segmentation using gabor filters," in *90CH2930-6/90/0000-0014*, pp. 14–19.

[82] Z. Yan and L. Harold, "Texture segmentation by symmetric and assymetric filters," in *0-8186-6950-0/94*, pp. 630–634.

[83] N. Isaac, T. Tan, and J. Kittler, "On local linear transform and gabor filter representation of texture," in *0-8186-2920-7/92*, pp. 627–631, IEEE.

[84] J. Teti and H. Kritikos, "Sar ocean image decomposition using gabor expansion," in *Transactions geoscience and remote sensing*, vol. 30, pp. 192–196, IEEE.

[85] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," in *Transactions on Computing*, vol. c, pp. 90 – 93, IEEE, January 1974.

[86] W. Pennebaker, *Image data compression standard*. Renhold, New York, 1993.

[87] I.-M. Pao and M.-T. Sun, "Computation reduction for discrete cosine transform," in *Circuits and Systems, 1998. ISCAS '98. Proceedings of the 1998 IEEE International Symposium on*, vol. 4, pp. 285–288, IEEE, 1998.

[88] L. David and D. Nguyen, "Removal of subjective redundancy from dct-coded images," in *proceedings*, vol. 138, pp. 345–350, IEEE, October 1991.

[89] M. Unser, "Local linear transform for texture measurements," in *Signal Processing*, vol. 11, pp. 61–79, July 1986.

[90] D. Trainor, J. Heron, and R. Woods, "Implementation of the 2d dct using a xilinx xc6264 fpga," in *Signal Processing Systems, 1997. SIPS 97 - Design and Implementation., IEEE Workshop on*, pp. 541 –550, IEEE, 1997.

[91] T. Lee, "Image representation using 2d gabor wavelets," in *Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 959–971, IEEE, October 1996.

[92] B. Jordan and N. Nguyen, "Specific perimeter - a graininess parameter for formation and print-mottle textures," in *Paperi ja Puu–Paper och*, vol. Tra 6, pp. 476–483, July 1986.

[93] R. J. Trepanier, "User-friendly system analyzes paper formation, dirt speck content, and solid-print nonuniformity," in *Tappi Journal*, pp. 153–157, 1989.

[94] R. Trepanier, B. Jordan, and N. Nguyen, "Specific perimeter:a static for assessing formation and print quality by image analysis," in *Tappi Journal*, vol. 10, pp. 191–196.

[95] A. J. Danker and A. Rosenfeld, "Blob detection by relaxation," in *Transactions on Pattern Analysis and Machine Intelligence*, vol. 3, pp. 79–92, IEEE, 1981.

[96] B. Mandelbrot, *The fractal Geometry of Nature*. San Fransisco:freeman, 1982.

[97] Voss, *Random Fractals:Characterisation and measurement, in scaling phenomena in disordered systems*. R. Pynn and A. Skjeltorp, Eds., Plenum, New York, 1988.

[98] S. Peleg, N. Hartley, and D. Anvir, "Multiple resolution texture analysis and classification," in *Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 518–523, IEEE, 1984.

[99] B. Chaudhuri and N. Sarkar, "Texture segmentation using fractal dimension," in *Pattern Recognition letters*, vol. 17, pp. 72–77, IEEE, January 1995.

[100] F. Albregtsen, B. Nielsen, and K. Yogesan, "Fractal dimension, only a fraction of the truth?," in *Pattern Recognition,Conference C: Image, Speech and Signal Analysis, Proceedings., 11th IAPR International Conference on*, vol. III, pp. 733 –736, 1992.

[101] A. Pentland, "Fractal-based description of natural scenes," in *Transactions On pattern analysis and machine intelligence*, vol. PAMI-6, pp. 661–674, IEEE, 1984.

[102] Y. Liu and Y. Li, "Image feature extraction and segmentation using fractal dimension," in *International conference on information, communications and signal processing, Singapore,*, pp. 975–979, IEEE, September 1997.

[103] T. Higuchi, "Approach to an irregular time series on the basis of the fractal theory," in *Physica D*, vol. 31, pp. 277–283, 1988.

[104] M. Katz, "Fractal and the analysis of waveforms," in *Comput. Biol Med*, vol. 18, pp. 145–156, 1998.

[105] A. Accardo, M. Affinito, M. Carrozzi, and F. Bouquet, "Use of fractal dimension for the analysis of electroencephalographic time series," in *Biol. Cybern.*, vol. 77, pp. 339–350, 1997.

[106] A. Pentland, "Shading into texture," in *Artificial intelligence*, vol. 29, pp. 147–170, 1986.

[107] R. Esteller, G. Vachtseranos, J. Echauz, and B. Litt, "A comparison of fractal dimension algorithms using synthetic and experimental data," in *0-7803-5471-0/99*, pp. III–199—III–202, 1999.

[108] N. Sarkar and B. Chaudhuri, "An efficient approach to estimate fractal dimension of textural images," in *Pattern Recognition letters*, vol. 25, pp. 1035–1041, 1992.

[109] A. Penn, "Fractal dimension of low-resolution medical images," in *18th International conference of the IEEE Engineering in Medicine and Biology Society, Amsterdam*, pp. 1163–1165, 1996.

[110] F. Arduini, S. Fioravanti, D. Giusto, and F. Inzirillo, "Multifractals and texture classification," in *Image Processing and its Applications*, pp. 454–457, 1992.

[111] A. Conci and C. Proenca, "A box-counting approach to color segmentation," in *Image Processing, 1997. Proceedings., International Conference on*, vol. 1, pp. 228 –230, 1997.

[112] J. Keller and S. Chen, "Texture description and segmentation through fractal geometry," in *Computer Vision, Graphics and Image Processing*, vol. 45, pp. 150–166, 1989.

[113] A. Kouzani, F. He, and K. Sammut, "Face image matching using fractal dimension," in *image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on*, vol. 3, pp. 642 –646, 1999.

[114] H. Kaneko, "A generalized fractal dimension and its application to texture analysis-fractal matrix model," in *Acoustics, Speech, and Signal Processing*, vol. 3, pp. 1711–1714, May 1989.

[115] J. Kux and G. Henebry, "Multi-scale texture in sar imagery: Landscape dynamics of the pantanal, brazil," in *Electronic Proceedings of IGARSS '94*, pp. 359–1364, 1995.

[116] L. Jacques, "About lacunarity, some links between fractal and integral geometry, and an application to texture segmentation," in *CH2934-8/90/0000/0380*, pp. 380–384, August 1990.

[117] A. Niranjan and M. Sunanda, "Analysis of texture images using robust fractal description," in *0-8186-6250-6/94*, June 1994.

[118] S. Haykin, *Neural Networks*. US Imports PHIPEs, 1998.

[119] D. Golberg, *Genetic Algorithms in Search Optimisation and Machine Learning*. Addisions-Wesley, 1989.

[120] A. Nandi and L. Jack, "Genetic algorithms for feature selection in machine condition monitoring with vibration signals," in *Vision, Image and Signal Processing, IEE Proceedings*, vol. 147, pp. 205 –212, IEE, June 2000.

[121] S. Reeves and Z. Zhe, "Sequential algorithms for observation selection," in *Transactions on Signal Processing*, vol. 47, pp. 123 –132, IEEE, January 1999.

[122] S. Reeves, "An improved sequential backward selection algorithm for large-scale observation selection problems,"

[123] A. Jain and Zongker, "Feature selection: Evaluation, application, and small performance," in *Transactions on pattern Analysis and Machine Intelligence*, vol. 19, pp. 153–158, IEEE, February 1997.

[124] A. Jain, R. Duin, , and J. Mao, "Statistical pattern recognition:a review," in *Transactions on pattern Analysis and Machine Intelligence*, p. 4.

[125] P. Pudil, J. Novovicova, and K. J., "Floating search methods in feature selection," in *Pettern recognition Letters*, vol. 15, pp. 1119–1125, November 1994.

[126] M. Unser and E. Murray, "Nonlinear operators for improving texture segmentation based on features extracted by spatial filtering," in *Transactions on Systems, Man and Cybernetics*, vol. 20, pp. 805–815, IEEE.

[127] F. Ade, "Characterisation of textures by eigenfilters," in *Signal Processing*, vol. 5, pp. 451–457, 1983.

[128] F. Pedersen, M. Bergstrom, E. Bengtsson, and E. Maripuu, "Principal component analysis of dynamic pet and gamma camera images: a methodology to visualize the signals in the presence of large noise," in *Nuclear Science Symposium and Medical Imaging Conference, 1993., 1993 IEEE Conference Record*, vol. 3, pp. 1734 –1738, IEEE, November 1993.

[129] M. Unser and F. Ade, "Feature extraction and decision procedure for automated inspection of textured materials," in *Pattern Recognition Letters*, vol. 2, pp. 185–191, March 1984.

[130] S. Baronti, R. Carla, S. Sigismondi, and L. Alparone, "Principal component analysis for change detection on polarimetric multitemporal sar data," in *Geoscience and Remote Sensing Symposium, 1994. IGARSS '94. Surface and Atmospheric Remote Sensing: Technologies, Data Analysis and Interpretation., International*, vol. 4, pp. 2152 –2154, IEEE, August 1994.

[131] O. Faugeras, "Texture analysis and classification using a human visual model," in *I.R.I.A., Institut de Recherche d'Informatique et d'Automatique*, pp. 549–553, 78150 Le Chesnay, France.

[132] K. Laws, "Texture energy measures," in *Proceedings of Image understanding Workshop*, pp. 47–51, November 1979.

[133] R. Duda and D. Stork, *Pattern Classification and Scene Analysis Part 1: Pattern Classification*. John Wiley and Sons, 2000.

[134] C. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[135] D. Rumelhart, R. Durbin, and Y. Chauvin, *Backpropagation: Theory, Arhitectures, and Applications*. Hillsdale, NJ:Lawrence Erlbaum.

[136] J. Aliana, T. Craig, and M. Robert, *Hand book of neural computing applications*. Academic Press.

[137] P. Brodatz, *Texture: a photographic album for artists and designers*. Dover, New York, 1966.

[138] J. Kitter, M. Hatef, R. Duin, and J. Matas, "On combining classifiers," in *Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 226 –239, IEEE, March 1998.

[139] L. Tarassenko, *Guide to Neural Computing Applications*. Butterworth-Heinemann, 1998.

[140] L. A. Zadeh., "Fuzzy sets," in *Information and Control*, vol. 8, pp. 338–353, 1965.

[141] C. Vassilios, A. Bors, and I. Pitas, "Multimodal decision-level fusion for person authentification," in *Transactions on Systems, Man, and Cybernetics*, vol. 29, pp. III–199—III–202, IEEE, November 1999.

[142] D. Cabaniss, Z. Cason, L. Lemos, and H. Benghuzzi, "The assessment of an endocervical component in cervicovaginal smears with the papnet system," in *Biomedical Engineering Conference*, pp. 357–361, IEEE roceedings of the 1997 Sixteenth Southern, April 1997.

[143] T.-M. Lee, G. Awcock, A. Dilley, and P. Anglim, "An investigation into the standardisation of the specific perimeter measure for the quantification of the quality in paper coating uniformity assessment," in *Le Creusot France*, vol. 67, QCAV, May 2001.